UNIVERSIDADE FEDERAL DE UBERLÂNDIA

Breno Palma Miele Aniceto

Um Estudo Baseado em Aprendizado de Máquina e SHAP para Detecção de Ataques DDoS utilizando o Dataset CICDDoS2019

Uberlândia, Brasil

2025

UNIVERSIDADE FEDERAL DE UBERLÂNDIA

Breno Palma Miele Aniceto

Um Estudo Baseado em Aprendizado de Máquina e SHAP para Detecção de Ataques DDoS utilizando o Dataset CICDDoS2019

Trabalho de conclusão de curso apresentado à Faculdade de Computação da Universidade Federal de Uberlândia, como parte dos requisitos exigidos para a obtenção título de Bacharel em Sistemas de Informação.

Orientador: Diego Nunes Molinos

Universidade Federal de Uberlândia – UFU

Faculdade de Computação

Bacharelado em Sistemas de Informação

Uberlândia, Brasil

2025

Breno Palma Miele Aniceto

Um Estudo Baseado em Aprendizado de Máquina e SHAP para Detecção de Ataques DDoS utilizando o Dataset CICDDoS2019

Trabalho de conclusão de curso apresentado à Faculdade de Computação da Universidade Federal de Uberlândia, como parte dos requisitos exigidos para a obtenção título de Bacharel em Sistemas de Informação.

Trabalho aprovado. Uberlândia, Brasil, 14 de maio de 2025:

Diego Nunes Molinos
Orientador

Rodrigo Sanches Miani

Fernanda Maria da Cunha Santos

Uberlândia, Brasil
2025

Dedico este trabalho aos meus pais e à minha família, pelo apoio incondicional e pela paciência ao longo desta jornada acadêmica, e aos meus amigos, que estiveram ao meu lado, incentivando-me em cada etapa deste projeto.

Agradecimentos

Agradeço ao meu orientador Diego Nunes Molinos, por sua paciência, orientação e valiosas contribuições que foram fundamentais para a realização deste trabalho. Agradeço também aos professores do curso, pelo conhecimento compartilhado, e aos colegas de turma, pelo companheirismo e apoio mútuo durante esta jornada.

Resumo

Este trabalho propôs um mecanismo de detecção de ataques DDoS baseado em aprendizado de máquina, utilizando o dataset CICDDoS2019, com foco na análise de fluxo de rede e na redução da complexidade dos modelos. Foram avaliados cinco algoritmos supervisionados (Decision Tree, C4.5, Random Forest, XGBoost e MLP) por meio de métricas como F1-Score, AUC e análise de explicabilidade com SHAP. Após pré-processamento, que incluiu remoção de features redundantes e tratamento de desbalanceamento, os modelos baseados em árvores (Random Forest e XGBoost) alcançaram os melhores resultados, com F1-Score de 0,9995 e AUC próximo de 1,0. A análise SHAP revelou que features como `ACK Flag Count` e `Fwd Packet Length Min` são as mais influentes na detecção de tráfego malicioso. Conclui-se que o Random Forest é o modelo mais eficiente, equilibrando desempenho, estabilidade e interpretabilidade, enquanto o XGBoost, embora mais preciso, apresenta maior complexidade devido à abordagem binária adotada, que abrange diferentes tipos de ataques DDoS.

Palavras-chave: Aprendizado de Máquina; Detecção de Ataques DDoS; Segurança de Redes; SHAP; Inteligência Artificial Explicável; CIC-DDoS2019

Lista de ilustrações

Lista de tabelas

Lista de abreviaturas e siglas

AC	Acurácia
AUC	Área Sob a Curva ROC (*Area Under the ROC Curve*)
CERT.br	Centro de Estudos, Resposta e Tratamento de Incidentes de Segurança no Brasil
CSV	*Comma-Separated Values*: Formato de arquivo tabular amplamente utilizado para armazenar dados estruturados, no qual os valores são separados por vírgulas.
CICFlowMeter	Ferramenta para Extração de Características de Tráfego de Rede
DDoS	Negação de Serviço Distribuída (*Distributed Denial of Service*)
DoS	Negação de Serviço (*Denial of Service*)
DNN	Redes Neurais Profundas (*Deep Neural Networks*)
FN	Falso Negativo (*False Negative*)
FP	Falso Positivo (*False Positive*)
IA	Inteligência Artificial
IDS	Sistema de Detecção de Intrusões (*Intrusion Detection System*)
IoT	Internet das Coisas (*Internet of Things*)
ML	Aprendizado de Máquina (*Machine Learning*)
PR	Precisão (*Precision*)
PCC	Coeficiente de Correlação de Pearson (*Pearson Correlation Coefficient*)
RC	*Recall* (Revocação/Sensibilidade)
ROC	Curva de Característica de Operação do Receptor (*Receiver Operating Characteristic*)
SDN	Redes Definidas por Software (*Software-Defined Networking*)
SHAP	*SHapley Additive exPlanations* (Técnica de interpretabilidade de modelos de ML)

SVM	Máquina de Vetores de Suporte (*Support Vector Machine*)
TCP	Protocolo de Controle de Transmissão (*Transmission Control Protocol*)
USB-IDS	*Dataset* de Ataques do Sistema de Detecção de Intrusões da University of Skövde
VP	Verdadeiro Positivo (*True Positive*)
VN	Verdadeiro Negativo (*True Negative*)
XAI	Inteligência Artificial Explicável (*Explainable Artificial Intelligence*)

Sumário

1 Introdução

Os ataques distribuídos de negação de serviço, ou comumente chamados de DDoS, têm se tornado cada vez mais sofisticados e frequentes, representando uma ameaça significativa à segurança cibernética em todo o mundo. Ataques DDoS têm experimentado um crescimento significativo, afetando diversos setores em todo o mundo, no primeiro semestre de 2024 houve um aumento de 46% na quantidade de ataques deste tipo em relação ao mesmo peŕiodo no ano anterior (GCORE, 2024).

De acordo com Akamai (2023), os setores financeiro e de comércio são os mais visados, devido ao grande potencial de ganho financeiro, seja por meio de extorsões ou causando prejuízos diretos às vítimas. Esses ataques sobrecarregam redes e servidores com um volume massivo de requisições maliciosas, visando tornar sistemas e serviços indisponíveis, impactando diretamente operações essenciais (AKAMAI, 2023; KAUR; KRISHNA; PATIL, 2025).

Conforme Kaur, Krishna e Patil (2025), apenas no primeiro trimestre de 2024, foram contidos aproximadamente 4,5 milhões de ataques DDoS, com aumento de 80% nos ataques baseados em DNS em relação ao período anterior. Países como a Suécia registraram crescimento de 466% no número de incidentes, enquanto setores altamente expostos à internet, como criptomoedas, jogos e apostas, lideraram o volume de alvos no quarto trimestre de 2024.

Adicionalmente, eventos de grande repercussão, como o ataque de 3,47 Tbps à Microsoft (2021), 2,3 Tbps à Amazon AWS (2020) e 1,35 Tbps ao GitHub (2018) demonstram que mesmo organizações com infraestrutura robusta são vulneráveis a esse tipo de ofensiva. Esses dados evidenciam a urgência no desenvolvimento de estratégias eficientes de detecção e mitigação (KAUR; KRISHNA; PATIL, 2025).

Em um ambiente altamente conectado, garantir a proteção contra DDoS é vital para a preservação da continuidade dos negócios e a segurança de infraestruturas críticas. Importante ressaltar que, a complexidade crescente desses ataques tem se tornado um desafio para o campo da segurança cibernética (G2, 2024; GCORE, 2024). Dados da (NETSCOUT, 2023) revelam que menos da metade dos ataques DDoS são mitigados com sucesso, o que destaca a necessidade de desenvolver soluções mais eficazes. Os usuários maliciosos empregam cada vez mais técnicas avançadas baseadas em Inteligência Artificial (IA), tornando a detecção de tráfego malicioso um desafio cada vez maior (G2, 2024; GCORE, 2024).

Nesse contexto, a utilização de técnicas de aprendizado de máquina para a mitigação desses ataques surge como uma estratégia promissora, permitindo a detecção e

resposta mais eficaz às ameaças em tempo real. A mitigação de ataques DDoS, especialmente com o uso de técnicas de aprendizado de máquina e a racionalização dos modelos (IA explicável) com objetivo de otimizar a operação desses modelos, é um tema de extrema importância, considerando a crescente complexidade e frequência desses ataques.

1.1 Hipótese

A utilização de algoritmos de aprendizado de máquina supervisionado, combinados com técnicas de análise de explicabilidade como SHAP, possibilita a construção de modelos capazes de detectar tráfego malicioso relacionado a ataques DDoS com alto desempenho, ao mesmo tempo em que reduz a complexidade do modelo por meio da identificação e seleção das *features* mais relevantes, tornando o sistema mais eficiente e interpretável para aplicações em ambientes reais de rede.

1.2 Objetivos

Diante do contexto apresentado, o objetivo deste trabalho é implementar um mecanismo de detecção de ataques DDoS baseado em Aprendizado de máquina, com foco na otimização, racionalização do processo de análise de fluxo rede. Este mecanismo de detecção se diferencia dos apresentados na literatura por realizar uma análise compreensiva do conjunto de *features* utilizado com o intuito de reduzir a complexidade do modelo proposto. Isso sugere um modelo mais eficiente para a classificação de tráfego malicioso.

1.2.1 Objetivos Específicos

- Analisar os principais algoritmos de aprendizado de máquina supervisionado aplicados à detecção de ataques DDoS,
- Pré-processar o conjunto de dados CICDDoS2019, realizando limpeza, transformação e seleção de atributos relevantes, a fim de reduzir redundâncias e aprimorar a eficiência dos modelos de classificação,
- Avaliar o desempenho de diferentes modelos de aprendizado de máquina com base em métricas como F1-Score, AUC e matriz de confusão, utilizando validação cruzada e abordagem binária de classificação,
- Aplicar técnicas de interpretabilidade com SHAP para identificar as features mais influentes na classificação do tráfego e comparar a complexidade explicativa entre os modelos.

1.3 Divisão da Monografia

Esta monografia está organizada em seis capítulos, estruturados de forma a conduzir o leitor da contextualização teórica à aplicação prática e análise dos resultados. O Capítulo 1 apresenta a introdução ao tema, contextualizando os ataques de negação de serviço distribuído (DDoS), sua evolução, impacto e a motivação para o uso de técnicas de aprendizado de máquina na mitigação dessas ameaças. O Capítulo 2 trata da fundamentação teórica, abordando os conceitos essenciais sobre ataques DDoS, aprendizado de máquina, suas categorias, métricas de avaliação e exemplos práticos aplicados à cibersegurança. O Capítulo 3 reúne os trabalhos relacionados, com destaque para estudos que aplicam inteligência artificial explicável (XAI) e algoritmos supervisionados na detecção de tráfego malicioso em redes modernas, especialmente em ambientes SDN. O Capítulo 4 descreve a metodologia adotada, incluindo a percepção do problema, seleção do dataset CICDDoS2019, escolha dos algoritmos, processo de pré-processamento e critérios de avaliação dos modelos. O Capítulo 5 apresenta o desenvolvimento experimental e os resultados obtidos, com ênfase na análise de desempenho dos algoritmos, curvas de aprendizado, validação cruzada e análise de explicabilidade por meio da técnica SHAP. Por fim, o Capítulo 6 contempla as conclusões do estudo, destacando as contribuições alcançadas e apontando direções para trabalhos futuros.

2 Fundamentação Teórica

Este capítulo apresenta os conceitos fundamentais que embasam o estudo sobre a detecção de ataques DDoS utilizando aprendizado de máquina e o framework SHAP. Inicialmente, discute-se os ataques de Negação de Serviço (DoS) e Negação de Serviço Distribuído (DDoS), explorando sua definição, histórico, classificações e implicações, com ênfase em casos emblemáticos e avanços recentes em estratégias de mitigação. Em seguida, aborda-se o uso de aprendizado de máquina na classificação de ataques DDoS, incluindo os principais tipos de aprendizado, sua aplicação na detecção de anomalias e as métricas de avaliação de desempenho dos classificadores. Esses tópicos são essenciais para compreender o contexto do problema e as bases teóricas que sustentam a metodologia e os resultados apresentados nos capítulos subsequentes.

2.1 Ataques de Negação de Serviço

Os ataques de Negação de Serviço (DoS) e Negação de Serviço Distribuído (DDoS) buscam interromper ou degradar significativamente o funcionamento de servidores, serviços ou redes, sobrecarregando-os com um volume de tráfego intenso. Um ataque DoS tradicional parte de uma única origem, enquanto um ataque DDoS utiliza múltiplos dispositivos, geralmente coordenados em uma *botnet*, para gerar simultaneamente um grande volume de requisições (ZARGAR; JOSHI; TIPPER, 2013). Essa coordenação provoca o esgotamento dos recursos do sistema alvo, dificultando o acesso de usuários legítimos ao serviço (ZARGAR; JOSHI; TIPPER, 2013).

De acordo com (ZARGAR; JOSHI; TIPPER, 2013), ataques DDoS podem ser classificados conforme a natureza do ataque, e podem ser divididos nas seguintes categorias:

1. Ataques Volumétricos: Focados em consumir a largura de banda da rede do alvo.
2. Ataques de Protocolo: Explorando vulnerabilidades nas camadas de rede e transporte, esgotam recursos de dispositivos de infraestrutura.
3. Ataques de Aplicação: Projetados para esgotar os recursos específicos de uma aplicação, como processamento e memória.

2.1.1 Histórico

Ataques DDoS remontam aos anos 1990, época em que surgiram os primeiros *malwares* e redes de *botnets* (OSTERWEIL; STAVROU; ZHANG, 2019). Um dos pri-

meiros grandes incidentes ocorreu em 2000, quando ataques DDoS interromperam o funcionamento de portais como Yahoo, Amazon e eBay (VILLAçA, 2018). Esses ataques mostraram o potencial destrutivo da técnica e resultaram em perdas financeiras consideráveis.

A partir da década de 2010, com o crescimento da Internet das Coisas (IoT) dispositivos IoT inseguros passaram a ser alvo fácil para *botnets* massivas (Cloudflare, 2023). Ataques como a Mirai Botnet em 2016, causou interrupções significativas em grandes plataformas (ZEIFMAN; BEKERMAN; HERZBERG, 2016), ilustrando a capacidade de ataques distribuídos em escala global. (Cloudflare, 2023).

Entre 2020 e 2024, houve um aumento significativo nos ataques DDoS, com relatórios indicando um crescimento de até 108% em 2024 em relação a 2023, segundo dados da StormWall (STORMWALL, 2025). Além disso, a Cloudflare registrou um aumento de 53% nos ataques mitigados em 2024 em comparação com 2023, totalizando 21,3 milhões de ataques (CLOUDFLARE, 2025). Esses números refletem a crescente sofisticação e frequência dos ataques, impulsionados pelo crescimento de botnets e pela automação de processos de ataque, conforme analisado por F5 Labs (LABS, 2024), que atribui o aumento a fatores como instabilidade geopolítica e vulnerabilidades exploradas. Outros estudos, como o relatório da Gcore, corroboram essa tendência, destacando um aumento de 46% no primeiro semestre de 2024 em relação ao mesmo período de 2023 (GCORE, 2024)

2.1.2 Avanços Recentes na Área de Detecção e Mitigação de Ataques DDoS

A evolução dos ataques DDoS motivou novas estratégias de defesa, incluindo: (a) Inteligência Artificial e aprendizado de máquina, que identificam automaticamente padrões anômalos e distinguem o tráfego legítimo do malicioso, permitindo respostas rápidas e eficazes, (b) Escalabilidade em Nuvem, com soluções como AWS Shield e Cloudflare, capazes de absorver grandes volumes de tráfego hostil preservando a operação normal dos serviços; e (c) Redundância e Balanceamento de Carga, distribuindo o tráfego entre múltiplas rotas para evitar sobrecargas localizadas (ZARGAR; JOSHI; TIPPER, 2013).

Essas estratégias representam avanços fundamentais no combate a ataques DDoS, considerando que as abordagens tradicionais já não são suficientes para enfrentar a complexidade e escala dos ataques atuais. O uso de Inteligência Artificial permite uma resposta mais precisa e ágil, detectando ataques ainda nos estágios iniciais. Da mesma forma, soluções baseadas em nuvem garantem maior resistência devido à capacidade de adaptação dinâmica frente a grandes volumes de tráfego hostil. Além disso, a implementação da redundância e balanceamento de carga fortalece a resiliência das infraestruturas, minimizando os impactos causados por ataques concentrados em pontos críticos específicos (SAHOSH et al., 2024).

2.1.3 Ataques DDoS e suas Implicações

O ataque utilizando a Mirai Botnet, ocorrido em 2016, tornou-se um marco significativo devido à sua escala inédita. Explorando vulnerabilidades de segurança em dispositivos conectados à Internet das Coisas (IoT), como câmeras IP e roteadores domésticos, o *malware* Mirai infectou milhares de dispositivos que posteriormente foram usados para lançar um ataque DDoS massivo contra o provedor de DNS Dyn. Este ataque resultou em interrupções generalizadas de serviços populares como Twitter, Netflix e GitHub, destacando a vulnerabilidade crítica representada pelos dispositivos IoT mal protegidos e impulsionando iniciativas globais por maior segurança nesses equipamentos (ZEIFMAN; BEKERMAN; HERZBERG, 2016).

O ataque DDoS contra o GitHub em 2018 representa um dos mais intensos já registrados até aquele momento, atingindo um pico impressionante de 1,35 terabits por segundo de tráfego malicioso. Este ataque tentou saturar os recursos da infraestrutura do GitHub, ameaçando seriamente sua operação e disponibilidade. Entretanto, a rápida mitigação realizada através da utilização de uma rede robusta de distribuição de conteúdo (CDN) permitiu absorver e distribuir o tráfego excessivo, demonstrando a eficácia de sistemas preventivos modernos na defesa contra ataques de larga escala (A10 Networks, 2020).

Os ataques DDoS sofridos pela Estônia em 2007 são amplamente considerados um ponto de virada na história dos conflitos cibernéticos internacionais. Com origens atribuídas a tensões políticas envolvendo a remoção de um monumento soviético na capital Tallinn, esses ataques tiveram como alvo serviços críticos governamentais, financeiros e de mídia, deixando-os indisponíveis por semanas. O incidente evidenciou a capacidade destrutiva de ataques cibernéticos direcionados contra infraestruturas nacionais e resultou em uma mudança significativa nas políticas globais sobre defesa cibernética, levando à criação de iniciativas como o Centro de Excelência em Defesa Cibernética da OTAN, em Tallinn (OTTIS, 2008).

2.1.4 Classificações de Ataques DDoS

De acordo com (DOULIGERIS; MITROKOTSA, 2004), os ataques de negação de serviço distribuídos podem ser categorizados pelas seguintes características:

- Vetor de ataque:
    1. Inundação: Grandes volumes de tráfego esgotam os recursos da vítima.
    2. Amplificação: Requisições enviadas com o IP da vítima geram tráfego não solicitado, sobrecarregando a vítima.

3. Exploração de Protocolo: Aproveitam-se de falhas em protocolos para consumir recursos.
4. Pacote Malformado: Envio de pacotes defeituosos para colapsar o serviço.

- Dinâmica da Taxa de Ataque:

1. Taxa Contínua: Mantém um tráfego constante durante toda a duração.
2. Taxa Variável: Comportamento do tráfego altera-se ao longo do ataque.

- Grau de automação:

1. Manuais: O atacante infecta dispositivos manualmente, enviando comandos para iniciar o ataque.
2. Semi-Automáticos: Buscas automáticas detectam dispositivos vulneráveis, mas ainda dependem de comandos manuais para iniciar o ataque.
3. Automáticos: Totalmente automatizados, com detalhes do ataque definidos no malware. Classificação por Vulnerabilidade Explorada

2.2 Aprendizado de Máquina

2.2.1 Aprendizado de Máquina na Classificação de Ataques DDoS

O aprendizado de máquina, comumento chamado de *Machine Learning* - ML, trata-se de um subcampo da inteligência artificial (IA) que capacita sistemas automatizados a analisar grandes volumes de dados, identificar padrões e tomar decisões com mínima intervenção humana (FACELI et al., 2011). Em vez de programar explicitamente cada tarefa, os sistemas de ML utilizam algoritmos que se ajustam automaticamente com a experiência, aprimorando sua precisão à medida que são expostos a novos dados, conforme discutido em (FACELI et al., 2011).

De acordo com Barros (2016), Machado (2011), os algoritmos de aprendizado de máquina podem ser categorizados em três grupos principais: (a) *Aprendizado Supervisionado:* Este método utiliza dados rotulados para prever saídas. Algoritmos como Regressão Logística, Random Forest e Redes Neurais buscam estabelecer relações entre as variáveis de entrada e as saídas, identificando padrões nos dados fornecidos. O aprendizado supervisionado é comum em problemas de classificação e regressão, (b) *Aprendizado Não Supervisionado:* Os algoritmos analisam dados não rotulados, agrupando-os em conjuntos (clusters) com base em semelhanças de características. Exemplos incluem K-Means e DBSCAN, usados para tarefas de agrupamento e associação, permitindo uma melhor compreensão das interações entre os dados e, (c) *Aprendizado por Reforço:* Este modelo envolve um agente que opera em um ambiente e aprende a partir de ações e recompensas.

A cada ação, o agente recebe feedback que o ajuda a otimizar sua estratégia ao longo do tempo (MACHADO, 2011).

A detecção de anomalias é uma tarefa crítica em cibersegurança, onde algoritmos de ML são treinados para identificar padrões anômalos no tráfego de rede, distinguindo comportamentos maliciosos de atividades normais (RAFIQUE et al., 2024). O aprendizado profundo (deep learning), um subcampo do ML, usa redes neurais complexas para realizar análises detalhadas, facilitando a detecção precisa de ataques, conforme destacado por (RAFIQUE et al., 2024).

2.2.2 Métricas de Avaliação de Classificadores

A análise e avaliação de um classificador podem ser realizadas a partir de seus erros e acertos, categorizados como Verdadeiro Positivo (VP), Verdadeiro Negativo (VN), Falso Positivo (FP) e Falso Negativo (FN), conforme discutido por (SOKOLOVA; LAPALME, 2009). O caso de Verdadeiro Positivo ocorre quando o sistema classifica corretamente um exemplo da classe alvo, enquanto o Verdadeiro Negativo indica uma classificação correta para um exemplo que não pertence a essa classe. Já os Falsos Positivos ocorrem quando o sistema atribui incorretamente um exemplo à classe alvo, e os Falsos Negativos surgem quando um exemplo da classe alvo é classificado como não pertencente a ela. Com esses elementos, constrói-se a matriz de confusão, onde as colunas representam as classes previstas pelo modelo, enquanto as linhas representam as classes reais, como também abordado por (FACELI et al., 2011)

Segundo Sokolova e Lapalme (2009), é comumente utilizado as seguintes métricas para avaliação de desempenho dos modelos gerados para detecção de ataques DDoS com base na matriz de confusão: Acurácia (AC), Precisão (PR), *Recall* (RC), *F1-Score* e a Área Sob a Curva ROC (AUC).

A **Acurácia (AC)** representa a proporção de classificações corretas em relação ao total de previsões, medida pela fórmula:

$$AC = \frac{VP + VN}{VP + VN + FP + FN} \tag{2.1}$$

A **Precisão (PR)** avalia a proporção de verdadeiros positivos entre todos os exemplos classificados como positivos, calculada por:

$$PR = \frac{VP}{VP + FP} \tag{2.2}$$

O ***Recall* (RC)**, ou sensibilidade, mede a proporção de verdadeiros positivos identificados corretamente pelo modelo, expresso como:

$$RC = \frac{VP}{VP + FN} \tag{2.3}$$

O ***F1-Score*** fornece uma média harmônica entre precisão e *recall*, equilibrando a exatidão e a abrangência das previsões do modelo:

$$F1 = 2 \times \frac{PR \times RC}{PR + RC} \tag{2.4}$$

A **Área Sob a Curva ROC (AUC)** é uma métrica que avalia a capacidade do modelo de distinguir entre as classes positiva e negativa, independentemente do limiar de classificação. A curva ROC (*Receiver Operating Characteristic*) plota a taxa de verdadeiros positivos (RC) contra a taxa de falsos positivos (FP/(FP + VN)) para diferentes limiares de decisão. O valor da AUC varia de 0 a 1, onde 1 indica um classificador perfeito, capaz de separar completamente as classes, e 0,5 representa um desempenho equivalente ao de uma classificação aleatória (FAWCETT, 2006). Essa métrica é particularmente útil em cenários desbalanceados, como na detecção de ataques DDoS, pois fornece uma visão geral do desempenho do modelo sem depender de um limiar específico.

Essas métricas permitem avaliar o modelo com maior precisão, fornecendo uma análise mais robusta sobre seu desempenho ao identificar ataques DDoS.

3 Trabalhos Relacionados

Este capítulo apresenta trabalhos correlatos que dialogam com a proposta desta pesquisa, fornecendo subsídios teóricos e práticos para o seu desenvolvimento.

3.1 The Role of Explainable AI in Network Security: A Case Study on DDoS Detection

O trabalho Kalutharage et al. (2023), explorou o uso de inteligência artificial explicável (XAI) para a detecção de ataques DDoS em redes IoT, com foco na identificação e explicação de anomalias de tráfego. Neste estudo os autores ofertaram um método que combina aprendizado de máquina com técnicas de explicabilidade para identificar os recursos mais influentes na detecção de ataques. A metodologia presente no trabalho envolveu a extração de características do tráfego com o uso da ferramenta CICFlowMeter, que analisou tráfego em três categorias: volumétrico, exaustão de estado TCP e baseado em camada de aplicação. Um modelo leve foi treinado com dados benignos e validado com ataques simulados da base de dados da *Benevento Instrution Detection System* (USB-IDS). Os resultados mostraram que o método proposto foi superior a técnicas tradicionais, como Random Forest e Decision Tree, em termos de precisão e eficiência. Ele alcançou 100% de precisão em detecções como o ataque Hulk Evasive, com explicações claras sobre quais características foram decisivas, como o número de pacotes por segundo e o comprimento de pacotes de tráfego. Além disso, o modelo foi validado em dispositivos de baixo desempenho, como o Raspberry Pi, demonstrando viabilidade para aplicações práticas.

Este trabalho é relevante para esta proposta pois visa integrar a explicabilidade ao processo de detecção, permitindo maior confiança nas decisões e reduzindo falsos positivos. Ele destaca a importância de métodos XAI para melhorar a segurança de redes IoT, fornecendo insights interpretáveis para mitigar ataques.

3.2 Machine learning algorithms to detect DDoS attacks in SDN

No trabalho apresentado por Santos et al. (2019), os autores investigaram o uso de aprendizado de máquina para detecção de ataques DDoS em Redes Definidas por Software (SDN), com foco na análise comparativa de algoritmos supervisionados. O objetivo principal do estudo foi avaliar a eficácia de modelos como Random Forest, Logistic Regression e Gradient Boosting na classificação de tráfego malicioso e legítimo em redes SDN. A me-

todologia incluiu a captura de dados de tráfego na tabela de fluxos, com 20.000 amostras divididas entre tráfego normal e malicioso. Os algoritmos foram treinados com 70% dos dados e validados com 30%, utilizando técnicas para evitar sobreajustes, como o K-fold. A eficácia foi medida em termos de acurácia, enquanto a eficiência avaliou o tempo de processamento. Os resultados indicaram que o Random Forest teve a maior acurácia, próximo de 100%, enquanto o Decision Tree apresentou o menor tempo de processamento, sendo o mais eficiente para uso em SDN. Entre os ataques, os à tabela de fluxo foram os mais fáceis de classificar, enquanto os ao controlador apresentaram maior dificuldade devido à similaridade com tráfego legítimo. O estudo concluiu que o Decision Tree é ideal para cenários em tempo real, e sugere a futura aplicação de modelos online para melhor adaptação a novos ataques. Este trabalho se relaciona ao projeto ao fornecer compreensão sobre algoritmos supervisionados para mitigação de DDoS em redes modernas, destacando a aplicabilidade prática em cenários com infraestrutura avançada, como as SDNs.

3.3 Detection and mitigation of DDoS attacks in SDN: A comprehensive review, research challenges and future directions

Em Singh e Behal (2020), os autores realizaram uma revisão sistemática abrangente sobre técnicas de detecção e mitigação de ataques DDoS em Redes Definidas por Software (SDN), com foco especial no uso de aprendizado de máquina. O artigo destaca os principais métodos supervisionados aplicados na identificação de anomalias de tráfego, além de identificar desafios como a complexidade computacional e a falta de explicabilidade nos modelos. A revisão incluiu uma análise detalhada da arquitetura SDN, destacando suas vantagens sobre redes IP tradicionais, como a separação dos planos de controle e dados, que possibilita uma gestão centralizada e dinâmica. Foram avaliados estudos que propunham soluções de detecção e mitigação de ataques DDoS, utilizando métricas como entropia, aprendizado supervisionado e modelos híbridos. Os trabalhos selecionados foram organizados por categorias, e cada solução foi analisada em termos de métricas de desempenho, uso de datasets e aplicabilidade prática. Os autores identificaram que técnicas baseadas em aprendizado de máquina, como SVM e Random Forest, frequentemente oferecem maior precisão na classificação de tráfego malicioso. No entanto, apontaram limitações, como alta complexidade computacional e falta de explicabilidade nos modelos, que dificultam sua adoção em redes reais. Além disso, os métodos baseados em entropia foram destacados por sua leveza, embora apresentem dificuldades para diferenciar tráfego legítimo de ataques sofisticados.

A principal contribuição deste trabalho para esta proposta foi o mapeamento de lacunas na literatura, incluindo a necessidade de incorporar inteligência artificial explicável (XAI) e métodos que lidem com a heterogeneidade dos dados em redes complexas. Os

autores também ressaltaram a importância de desenvolver soluções escaláveis, capazes de operar em tempo real sem sobrecarregar os controladores SDN.

3.4 An entropy and machine learning based approach for DDoS attacks detection in software defined networks

O estudo conduzido por Hassan, Reheem e Guirguis (2024) propôs uma abordagem híbrida para a detecção e mitigação de ataques DDoS em Redes Definidas por Software (SDN), combinando métodos estatísticos baseados em entropia com técnicas de aprendizado de máquina. A fase estatística utilizou um mecanismo de detecção baseado em entropia, enquanto a fase de aprendizado de máquina empregou o algoritmo de clustering k-means para analisar o impacto dos usuários ativos na entropia do sistema. A arquitetura do modelo é composta por módulos de detecção baseados em entropia e aprendizado de máquina, integrados estrategicamente para aprimorar a eficácia na detecção e mitigação de ataques DDoS. A metodologia envolveu a pré-processamento dos dados, extração de características relevantes e agrupamento de fluxos de requisições em intervalos de tempo iguais. Após cada intervalo, o módulo de detecção por entropia avaliava a ocorrência de ataques e, em caso positivo, a técnica de aprendizado de máquina iniciava o clustering dos usuários ativos naquele intervalo em três clusters distintos: usuários normais, suspeitos e atacantes. Essa abordagem sequencial enfatiza a capacidade do modelo de responder dinamicamente a potenciais ataques, integrando detecção baseada em entropia e clustering de aprendizado de máquina para um mecanismo de defesa abrangente. A abordagem proposta foi experimentalmente avaliada utilizando três conjuntos de dados modernos: CIC-IDS2017, CSE-CIC-2018 e CICIDS2019. Os resultados demonstraram a eficácia do sistema em detectar e bloquear ataques súbitos e rápidos, destacando o potencial da abordagem para aprimorar significativamente a segurança contra ataques DDoS em ambientes SDN.

Este trabalho é relevante para esta proposta por integrar técnicas estatísticas e de aprendizado de máquina na detecção de ataques DDoS em SDN, oferecendo uma solução híbrida que combina a eficiência da detecção baseada em entropia com a capacidade de clustering do aprendizado de máquina. A utilização de datasets atualizados e a validação experimental reforçam a aplicabilidade prática da abordagem proposta.

3.5 Cybersecurity Defence Mechanism Against DDoS Attack with Explainability

O trabalho de Mahmood (2024), os autores propuseram um modelo de defesa para ataques DDoS na camada de aplicação (Camada 7), utilizando algoritmos de aprendi-

zado de máquina combinados com técnicas de explicabilidade. O sistema proposto opera em três estados: normal, observando e suspeito. Ao detectar um tráfego anormal, classificadores (Decision Tree, SVM, Logistic Regression, Naive Bayes e KNN) analisam os dados, seguidos por um classificador empilhado que aumenta a precisão da detecção. A técnica SHAP foi empregada para explicar as decisões, garantindo maior transparência. Em ataques contínuos, o sistema ativa automaticamente a mitigação, bloqueando usuários suspeitos. Os resultados obtidos comprovaram a eficácia e interpretabilidade da solução na distinção entre tráfego legítimo e malicioso.

3.6 Effective DDoS Mitigation via ML-Driven In-Network Traffic Shaping

Um paradigma inovador para a mitigação de ataques DDoS foi proposto por Zhao et al. (2024), por meio da introdução do conceito de *preference-driven and in-network enforced traffic shaping*. Em contraste às estratégias tradicionais, que buscam primordialmente identificar e filtrar exaustivamente o tráfego malicioso, essa abordagem enfatiza a priorização proativa do tráfego considerado desejado pela vítima, independentemente da estratégia utilizada pelo atacante. Para viabilizar essa proposta, os autores desenvolveram o sistema DFNet, que integra algoritmos de aprendizado de máquina com primitivas avançadas de plano de dados (*dataplane*), permitindo a codificação dinâmica das preferências do tráfego legítimo—definidas por complexos modelos de classificação—diretamente nos mecanismos de escalonamento e encaminhamento dos pacotes. Avaliações experimentais realizadas com enlaces de 80 Gbps e gargalo configurado em 40 Gbps demonstraram que o DFNet conseguiu garantir uma taxa de entrega de 99,93% do tráfego legítimo, mesmo sob ataques inéditos, com sobrecarga computacional inferior a 0,1%. Essa abordagem é especialmente relevante para esta proposta, pois oferece uma perspectiva complementar às técnicas baseadas em aprendizado de máquina investigadas neste trabalho, destacando-se pela capacidade de garantir a entrega rápida e eficiente do tráfego legítimo em situações críticas.

3.7 AE-MLP: A Hybrid Deep Learning Approach for DDoS Detection and Classification

No estudo conduzido por Wei et al. (2021), os autores apresentam o modelo AE-MLP, uma abordagem híbrida de aprendizado profundo que integra Autoencoders (AE) e Perceptrons Multicamadas (MLP) para a detecção e classificação de ataques DDoS. O AE é empregado para extrair características relevantes do tráfego de rede, enquanto o MLP realiza a classificação entre tráfego benigno e malicioso. Esta metodologia destaca-se pela

capacidade dos autoencoders de capturar padrões complexos e da MLP de fornecer classificações precisas. Publicado na *IEEE Access*, o trabalho reforça a aplicabilidade de modelos de aprendizado profundo em cenários de segurança cibernética, especialmente na detecção de DDoS. Sua relevância para esta pesquisa reside na demonstração de como técnicas avançadas de aprendizado de máquina podem ser utilizadas para melhorar a detecção de ataques, alinhando-se com o objetivo de desenvolver soluções robustas e interpretáveis.

3.8 A Machine Learning Approach for DDoS Detection on IoT Devices

No trabalho de Seifousadati et al. (2021), os autores exploram o uso de aprendizado de máquina para detectar ataques DDoS em dispositivos de Internet das Coisas (IoT). Dado o crescente número de dispositivos IoT e sua vulnerabilidade a ataques, esta pesquisa é crucial para garantir a segurança dessas redes. Os autores utilizam técnicas de aprendizado de máquina para analisar padrões de tráfego e identificar comportamentos anômalos que indiquem ataques DDoS. Disponível no *arXiv*, o estudo destaca a aplicabilidade do aprendizado de máquina em cenários de IoT, onde os recursos computacionais são limitados, mas a necessidade de segurança é alta. Sua relevância para esta pesquisa reside na demonstração de como modelos de aprendizado de máquina podem ser adaptados para ambientes de IoT, complementando as abordagens tradicionais de detecção de DDoS.

Tabela 1 – Comparação de trabalhos sobre detecção de DDoS com ML e XAI.

Trabalho	**Conjunto de dados**	**XAI**	**SDN**	**IoT**	**Modelo**
Kalutharage et al. (2023)	USB-IDS	Sim (SHAP)	Não	Sim	SAE
Souza et al. (2019)	SDN personalizado	Não	Sim	Não	Vários
Singh et al. (2020)	Revisão sistemática	Comenta sobre	Sim	Não	Vários
Hassan et al. (2024)	CIC-IDS2017, CSE-CIC-2018, CICD-DoS2019	Não	Sim	Não	Entropia + k-means
Mahmood e Avcı (2024)	CICDDoS 2019	Sim (SHAP)	Não	Não	Vários
Zhao et al. (2024)	Conjunto Personalizado	Não	Sim	Não	DFNet
Wei et al. (2021)	CICDDoS 2019	Não	Não	Não	AE-MLP
Seifousadati et al. (2021)	CICDDoS 2019	Não	Não	Sim	Vários
Este trabalho	CICDDoS 2019	Sim (SHAP)	Não	Não	RF, XGB, MLP, DT, C4.5

4 Método

O método adotado para o desenvolvimento deste trabalho seguiu uma sequência estruturada de etapas que visaram demonstrar a viabilidade e a eficácia de modelos baseados em aprendizado de máquina para a detecção e mitigação de ataques DDoS. A abordagem foi dividida em três macroetapas principais: percepção da problemática, modelagem e desenvolvimento do estudo, e validação e análise de resultados.

4.1 Percepção da Problemática

Nesta etapa buscou-se compreender a relevância e os desafios impostos pelos ataques DDoS por meio de um estudo sistemático da literatura. O estudo da literatura elucidou também soluções existentes, destacando especialmente abordagens baseadas em aprendizado de máquina para detectar tráfego malicioso, além de apontar limitações e lacunas ainda existentes nessas propostas. Essa análise sistemática fundamentou claramente a definição do objetivo central deste trabalho, direcionando-o para soluções mais inovadoras e eficazes nesse contexto.

4.2 Desenvolvimento e Avaliação dos Modelos

Esta etapa compreende a implementação prática da solução proposta, incluindo o preparo dos dados, a aplicação de algoritmos de aprendizado de máquina e a posterior análise dos resultados obtidos. Inicialmente, foram realizadas tarefas de pré-processamento, como a limpeza dos dados e a codificação das variáveis. Em seguida, múltiplos modelos supervisionados foram treinados e testados com o objetivo de identificar padrões de tráfego malicioso. Por fim, os modelos foram avaliados por meio de métricas de desempenho amplamente adotadas na literatura, permitindo verificar sua eficácia na detecção de ataques DDoS e embasar comparações entre diferentes abordagens.

4.2.1 Escolha do Conjunto de Dados (Dataset)

O conjunto de dados CICDDoS2019, disponibilizado pelo *Canadian Institute for Cybersecurity* (CIC) da Universidade de New Brunswick, foi selecionado para este trabalho. Este dataset é amplamente reconhecido pela comunidade acadêmica, sendo utilizado em estudos como (SEIFOUSADATI et al., 2021), (HASSAN; REHEEM; GUIRGUIS, 2024) e (WEI et al., 2021) devido à sua abrangência e qualidade, cobrindo uma ampla gama de tipos de ataques DDoS e cenários realistas de tráfego de rede.

A escolha desse dataset se justificou por vários motivos:

- **Diversidade dos dados**: Inclui amostras de tráfego legítimo e malicioso, simulando diferentes cenários de ataque.
- **Qualidade da anotação**: As etiquetas do conjunto de dados são detalhadas e confiáveis, permitindo um treinamento preciso dos modelos.
- **Relevância acadêmica**: Por ser amplamente citado em estudos na área, o dataset possibilita comparações diretas com outras abordagens presentes na literatura.

O *dataset* utilizado neste trabalho é uma versão pré-processada do *CIC-DDoS2019*, disponibilizada por Sharafaldin et al. (2022), contendo ajustes como remoção de duplicatas, tratamento de valores ausentes e correção de desbalanceamento, permitindo a reprodução dos experimentos com os mesmos dados utilizados.

4.2.2 Escolha dos Algoritmos de Inteligência Artificial

Os algoritmos de aprendizado de máquina supervisionado empregados neste estudo foram organizados em três categorias principais, selecionadas com base em sua ampla utilização na literatura e no desempenho consistente demonstrado em pesquisas voltadas à detecção de ataques DDoS. Essa classificação visa contemplar diferentes paradigmas de modelagem e níveis de complexidade, permitindo uma análise comparativa mais robusta e representativa.

- **Árvores de Decisão**: foram incluídas por sua natureza intuitiva e interpretabilidade, características que facilitam a compreensão do processo decisório do modelo. Esses algoritmos constroem estruturas hierárquicas baseadas em regras simples de decisão, o que os torna especialmente úteis em ambientes que demandam transparência e velocidade de inferência. Dois modelos dessa categoria foram utilizados, possibilitando avaliar sua eficiência em identificar padrões maliciosos com baixo custo computacional.
- **Modelos Ensemble**: foram escolhidos por sua reconhecida capacidade de combinar múltiplos classificadores fracos para formar modelos mais robustos e precisos. Essa abordagem permite mitigar problemas como o sobreajuste (overfitting), além de aumentar a generalização dos resultados. A presença de algoritmos ensemble no experimento foi essencial para investigar os benefícios advindos da agregação de modelos em contextos de tráfego de rede altamente variável.
- **Redes Neurais Profundas**: por sua vez, foram incluídas devido à sua habilidade em capturar relações não lineares complexas e extrair representações profundas dos

dados. Embora exijam maior poder computacional e cuidado no ajuste de hiperparâmetros, redes do tipo MLP (Multilayer Perceptron) têm se mostrado eficazes na detecção de anomalias em ambientes de rede. Um modelo representativo dessa classe foi implementado, com foco em arquiteturas consolidadas e bem documentadas, buscando avaliar seu desempenho frente às demais abordagens consideradas.

4.2.3 Pré-processamento do Dataset

O conjunto de dados passou por um processo de higienização e preparação, etapa fundamental para garantir a qualidade e a confiabilidade dos classificadores. Inicialmente, foram removidas *features* com baixa variabilidade ou impacto insignificante, como atributos com valores constantes ou redundantes, de forma a reduzir a dimensionalidade do conjunto de dados e evitar ruído durante o treinamento. Em seguida, a variável-alvo foi convertida para um formato binário, classificando os registros como tráfego *benigno* ou *maligno*, o que simplifica o problema de classificação e torna a avaliação dos modelos mais objetiva.

Além disso, foram tratados valores ausentes e inconsistências nos dados, assegurando a integridade do conjunto de treinamento e evitando viés ou falhas na aprendizagem dos algoritmos. Esses procedimentos são essenciais para aumentar a robustez dos modelos, melhorar o desempenho preditivo e garantir a validade dos resultados obtidos.

4.2.4 Treinamento dos Modelos

Os modelos foram treinados por meio de uma abordagem iterativa, com o objetivo de maximizar o desempenho na detecção de ataques DDoS. Para isso, o conjunto de dados foi dividido de forma estratificada em 70% para treinamento e 30% para teste, preservando a proporção entre classes. Diversos algoritmos foram avaliados com base em métricas padrão, como acurácia, precisão, revocação e F1-score, permitindo uma comparação objetiva entre os desempenhos.

Durante o treinamento, gráficos de evolução do F1-score foram utilizados para monitorar possíveis indícios de sobreajuste ou subajuste. Além disso, foram geradas matrizes de confusão e gráficos de comparação de erros residuais, oferecendo uma análise mais detalhada dos padrões de erro, especialmente em contextos com desbalanceamento de classes. Essas ferramentas foram fundamentais para compreender a capacidade de generalização de cada modelo diante da variabilidade do tráfego de rede. O processo foi estruturado em três etapas principais, descritas a seguir.

- **Baseline**: Inicialmente, foram construídos modelos simples com configurações padrão dos algoritmos, servindo como referência inicial para os experimentos. Esses

modelos estabeleceram um ponto de comparação para as versões otimizadas, permitindo avaliar os ganhos obtidos com ajustes de hiperparâmetros e estratégias de pré-processamento.

- **Divisão dos dados**: A divisão dos dados foi realizada utilizando a técnica *holdout*, com 70% dos registros destinados ao treinamento e 30% à avaliação. Para garantir a representatividade de ambas as classes (benigno e malicioso), essa divisão foi feita de forma estratificada. Além disso, aplicou-se a validação cruzada com $k = 5$ (*k-fold cross-validation*) durante o treinamento, o que permitiu reduzir a variância nos resultados e aumentar a confiabilidade das avaliações, especialmente em relação à generalização dos modelos.
- **Monitoramento**: Durante o treinamento, foram analisadas curvas de aprendizado, que demonstram a evolução do desempenho dos modelos em relação aos conjuntos de treino e teste. Essa análise possibilitou a identificação de comportamentos como *overfitting* (quando o modelo aprende excessivamente os dados de treino) ou *underfitting* (quando o modelo não consegue aprender padrões relevantes). Além disso, histogramas de distribuição dos erros e gráficos comparativos entre predições corretas e incorretas auxiliaram na compreensão do desempenho dos modelos frente à complexidade dos dados.

Os histogramas foram utilizados para analisar a distribuição dos erros residuais e acompanhar o ajuste dos modelos ao longo do treinamento, permitindo identificar desequilíbrios entre classes e avaliar a evolução da capacidade dos algoritmos em distinguir tráfego legítimo e malicioso.

4.3 Validação e Análise dos Resultados

Os modelos foram avaliados com base em métricas amplamente reconhecidas na literatura:

Tabela 2 – Métricas de avaliação utilizadas no estudo

Métrica	**Descrição**
Acurácia	Mede a proporção de predições corretas em relação ao total de instâncias avaliadas.
Precisão	Indica a porcentagem de predições positivas que estão corretas, refletindo a capacidade do modelo de evitar falsos positivos.
Recall (Sensibilidade)	Representa a proporção de instâncias positivas corretamente identificadas, medindo a capacidade do modelo de detectar ataques reais.
F1-score	Calculado como a média harmônica entre precisão e recall, proporcionando um equilíbrio entre essas duas métricas.
Matriz de confusão	Ferramenta para visualização da distribuição das classificações corretas e incorretas entre as diferentes classes.
AUC (Área sob a Curva ROC)	Avalia a capacidade do modelo de distinguir entre classes, sendo especialmente útil em situações com desbalanceamento de dados.

A aplicação do framework SHAP permitiu identificar as variáveis com maior impacto nas decisões dos modelos, facilitando a interpretação dos resultados e a comparação entre a complexidade e a transparência das diferentes abordagens adotadas.

4.3.1 Ferramentas Utilizadas

Para a execução do estudo, foram utilizadas as seguintes ferramentas:

- **Ambiente de desenvolvimento**: Python 3.10, utilizando o ambiente de notebooks do Kaggle, que oferece integração direta com o dataset e infraestrutura computacional de nuvem.
- **Bibliotecas de aprendizado de máquina**: Scikit-learn, TensorFlow e Keras.
- **Ferramentas de visualização**: Matplotlib e Seaborn para gráficos e visualizações de dados.
- **Análise de explicabilidade**: SHAP, para avaliar a transparência e interpretabilidade dos modelos gerados.
- **Conjunto de dados**: CICDDoS2019, acessado e gerenciado por meio da plataforma Kaggle.

4.3.2 Artefatos de Código

Esta seção descreve os artefatos digitais utilizados nos experimentos deste trabalho, incluindo o *dataset* pré-processado e o código implementado, disponíveis para consulta e reprodução, assegurando transparência e validação das etapas realizadas.

O código para este trabalho foi implementado em um notebook no Kaggle, intitulado "ML-SHAP", que utiliza os dados pré-processados para treinar modelos de aprendizado de máquina, realizar análise de explicabilidade com SHAP e gerar gráficos de avaliação. O notebook está disponível no Kaggle Palma (2025a) e no GitHub Palma (2025b).

5 Treinamento dos Modelos e Resultados

Este capítulo é dedicado ao treinamento e à otimização dos modelos de aprendizado de máquina aplicados à detecção de ataques DDoS. Nele, são apresentados os resultados parciais obtidos em cada etapa do processo, incluindo a preparação e análise exploratória dos dados, a seleção e o balanceamento dos atributos, o ajuste dos algoritmos, bem como a avaliação de desempenho dos modelos. Também são discutidos os aspectos relacionados à interpretabilidade das decisões, com o apoio de técnicas de explicabilidade.

5.1 Preparação e Higienização dos Dados

O conjunto de dados original *CIC-DDoS2019*, fornecido pelo *Canadian Institute for Cybersecurity* (CIC), consiste em arquivos no formato PCAP (*Packet Capture*) com aproximadamente 30 GB de capturas brutas de pacotes de rede, simulando diversos tipos de ataques DDoS e tráfego benigno. Esses dados estão organizados em conjuntos distintos de treinamento (*training*) e teste (*testing*), cada um associado a categorias específicas de ataque. Contudo, os modelos de aprendizado de máquina empregados neste trabalho não operam diretamente sobre os pacotes brutos, mas sim sobre *features* extraídas de fluxos de rede, que condensam informações relevantes de pacotes com atributos compartilhados, como endereços IP, portas e protocolos, tornando a análise mais eficiente e adequada à tarefa de classificação.

Para tornar os dados do *CIC-DDoS2019* utilizáveis em modelos de aprendizado de máquina, é empregada a ferramenta *CICFlowMeter*, que converte os pacotes PCAP em fluxos de rede bidirecionais e extrai mais de 80 *features* por fluxo, como duração, taxa de transferência, tamanho médio dos pacotes e contagem de *flags* TCP. Essas informações são essenciais para a identificação de padrões anômalos por Sistemas de Detecção de Intrusões (IDS). O próprio dataset já fornece uma versão processada em CSV, pronta para uso no treinamento dos modelos.

Para este trabalho, foi utilizada a versão reorganizada e limpa do *dataset* disponibilizado por (SHARAFALDIN et al., 2022). Essa versão parte dos dados em CSV fornecidos pelo *CIC-DDoS2019* e aplica correções adicionais, como a correção do desbalanceamento extremo das classes presente no *dataset* original, onde a proporção entre instâncias de ataque (*attack*) e benignas (*benign*) era de 4000:1. Para sanar esse problema, técnicas de reamostragem foram aplicadas, reduzindo a proporção para 9:1, o que permitiu um treinamento mais equilibrado e evitou que os modelos fossem enviesados em favor da classe majoritária (*malign*). Além disso foi realizada a remoção de 15% dos registros duplicados (aproximadamente 1,5 milhão de instâncias), o tratamento de valores ausentes e a limpeza

de atributos de *metadata*. Esses atributos de *metadata*, segundo estudo de D'hooge et al. (2022), podem introduzir viés em modelos de aprendizado de máquina, e sua remoção foi essencial para garantir a robustez dos modelos.

Após o *download* dos arquivos classificados como *Train* a partir do trabalho de (SHARAFALDIN et al., 2022), foi realizada a concatenação dos dados de treinamento em um único *DataFrame*, totalizando 121.676 instâncias, seguido da verificação da consistência das colunas (*features*) e dos rótulos (*labels*). A Figura 1 apresenta a proporção de exemplos de cada categoria presente nos dados originais concatenados, revelando um desbalanceamento entre as classes (aproximadamente 63% malicioso e 37% benigno). Esse desbalanceamento foi mantido nesta etapa inicial e tratado posteriormente durante o treinamento.

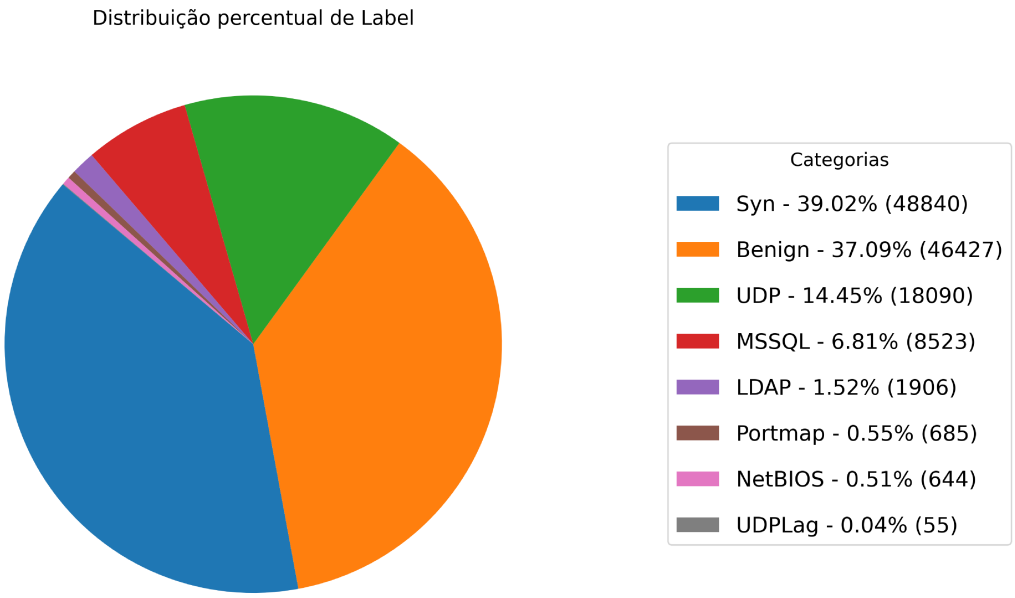

Figura 1 – Distribuição das classes no conjunto de dados concatenado

O tráfego Benign representa atividades normais, como navegação web, servindo como referência para detecção de anomalias. Syn (TCP-SYN Flood) é um ataque de protocolo que explora o handshake TCP, enquadrando-se no vetor de exploração de protocolo. UDP (UDP Flood) é um ataque volumétrico por inundação, MSSQL, LDAP, Portmap e NetBIOS são ataques volumétricos de amplificação, comuns em *botnets*, enquanto UDPLag, que causa atrasos na rede, pode ser classificado como ataque de protocolo ou pacote malformado. Essa diversidade de ataques reflete os vetores e categorias, justificando a escolha do dataset para treinar modelos de Aprendizado de Máquina e avaliar sua capacidade de detectar diferentes tipos de DDoS.

Foram também realizadas novas verificações para identificar duplicatas (aproximadamente 3% dos registros, ou 3.494 instâncias), as quais foram removidas. Embora o

dataset da plataforma Kaggle já apresentasse uma versão parcialmente limpa, uma nova inspeção foi conduzida para garantir a inexistência de valores nulos ou infinitos, os quais poderiam comprometer a robustez do modelo e das métricas de avaliação.

5.2 Análise Exploratória dos Dados (EDA)

As colunas foram classificadas automaticamente com base em seu tipo e número de valores distintos (*cardinalidade*). A classificação resultou nas seguintes categorias:

- **Categóricas**: colunas do tipo `object`, bem como colunas numéricas com baixa cardinalidade (menos de 10 valores distintos), que se comportam como categorias discretas.
- **Numéricas contínuas**: colunas numéricas com distribuição ampla e variada, excluindo as categóricas descritas acima.
- **Categóricas de alta cardinalidade**: colunas do tipo `object` com mais de 20 categorias únicas, tratadas separadamente devido à dificuldade de visualização e codificação.

Para cada grupo, foram utilizadas abordagens específicas de visualização:

- **Categóricas**: foram utilizados gráficos de barras (*countplots*) para representar a frequência de cada valor, separados por classe.
- **Numéricas contínuas**: foram aplicados *boxplots*, com o objetivo de visualizar a distribuição e a presença de valores extremos (outliers) por classe.
- **Candidatas numéricas com distribuição separável**: receberam tratamento com gráficos de densidade (*KDE plots*), permitindo observar a sobreposição (ou separabilidade) entre as classes.

A Figura 2, evidencia uma grande variabilidade dentro da classe `Benign`, mas também mostra que certos tipos de ataque, como `Syn` e `Portmap`, possuem distribuições bem concentradas com durações menores. Isso sugere que essa variável pode ser útil para distinguir entre comportamentos normais e maliciosos.

No gráfico de densidade da variável `Packet Length Mean`, ilustrado pela Figura 3, observa-se uma separação mais clara entre as classes: o tráfego benigno apresenta uma distribuição mais dispersa e centrada em valores positivos, enquanto ataques como `UDP` aparecem fortemente concentrados em valores baixos, reforçando o potencial discriminativo desse atributo.

Já o gráfico de frequência da variável `Protocol`, ilustrado pela Figura 4, mostra padrões interessantes: o protocolo com valor `6` (TCP) é dominante em ataques do tipo `Syn` e também no tráfego benigno, mas o protocolo `17` (UDP) aparece mais frequentemente associado a ataques como `Portmap` e `UDP-Lag`. Isso reforça a importância da variável `Protocol` na modelagem.

Ainda é possível observar *Outliers* nas variáveis `Flow Duration`, mas foram mantidos no conjunto, pois representam comportamentos típicos de ataques DDoS volumétricos, como fluxos extremamente curtos ou com duração nula.

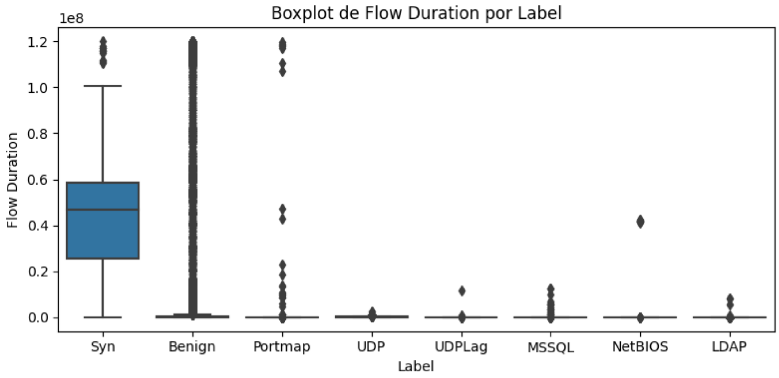

Figura 2 – Boxplot da variável `Flow Duration` por classe

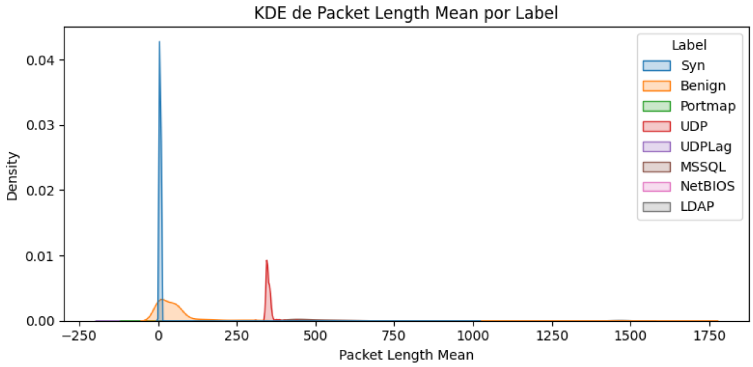

Figura 3 – Distribuição de densidade da variável `Packet Length Mean` por classe

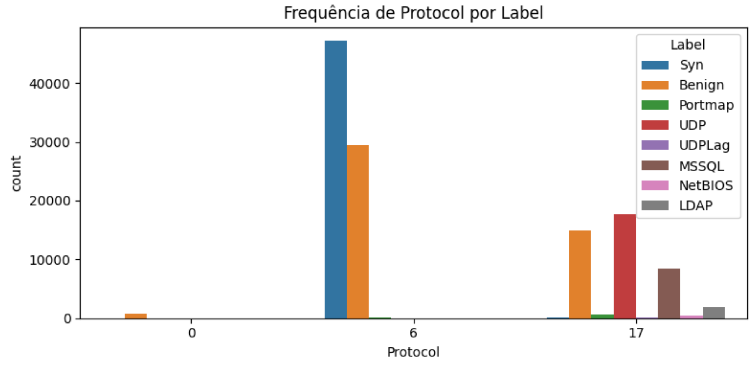

Figura 4 – Frequência da variável `Protocol` por classe

5.2.1 Seleção de Atributos

Nesta etapa foram removidas as variáveis que apresentavam o mesmo valor em todas as instâncias do conjunto de dados. Esse tipo de *feature* é comum em datasets de tráfego de rede e naturalmente não contribui para a diferenciação entre classes devido à ausência de variância. Com isso, o número de atributos foi reduzido de 78 para 66.

Em seguida, foi realizada uma análise de correlação entre os atributos numéricos com o objetivo de eliminar variáveis redundantes que pudessem introduzir colinearidade nos modelos. Para isso, utilizou-se o coeficiente de correlação de Pearson (PCC), que mede a relação linear entre pares de variáveis. Adotou-se como critério um limiar de 0,8. Quando esse valor era excedido, uma das variáveis do par correlacionado era removida, priorizando-se a manutenção daquela com maior variabilidade estatística. Essa abordagem segue recomendações como as descritas por (HAN et al., 2024), que destacam o uso do PCC como uma técnica eficaz de seleção de atributos ao remover características redundantes altamente correlacionadas, contribuindo para a redução da dimensionalidade e melhora da eficiência computacional do modelo.

Após essa filtragem, o conjunto de atributos foi reduzido para 32, mantendo apenas aquelas com maior capacidade discriminativa. Essa etapa foi essencial para tornar o treinamento mais eficiente, reduzindo a complexidade do modelo e evitando redundância nos dados.

5.3 Treinamento dos Modelos

Com o objetivo de avaliar o desempenho de diferentes abordagens de aprendizado de máquina, os cinco modelos escolhidos foram:

- **Árvores de Decisão**: `Decision Tree` e `C4.5` (implementado com entropia como critério de divisão);
- **Modelos Ensemble**: `Random Forest` e `XGBoost`;
- **Rede Neural**: `Multilayer Perceptron (MLPClassifier)` com duas camadas ocultas.

A Tabela 3 resume os modelos e suas configurações principais.

Tabela 3 – Modelos utilizados e parâmetros principais

Modelo	**Tipo**	**Parâmetros principais**
Decision Tree	Árvore de decisão	max_depth=10
C4.5 (Entropy)	Árvore de decisão	criterion='entropy', max_depth=10
Random Forest	Ensemble	n_estimators=100
XGBoost	Ensemble	n_estimators=100, eval_metric='logloss'
MLPClassifier	Rede neural	hidden_layers=(100, 50), max_iter=500

A variável-alvo `Label`, responsável por indicar o tipo de tráfego em cada registro, foi previamente convertida para um formato binário. Instâncias de tráfego legítimo foram rotuladas como `1`, enquanto qualquer tipo de ataque foi representado pela classe `0`. Essa transformação permitiu a aplicação direta de algoritmos de classificação supervisionada, simplificando o problema para uma tarefa de classificação binária.

A Figura 5 apresenta a distribuição das classes após a binarização, preservando o desbalanceamento original do conjunto de dados: aproximadamente 63% das instâncias representam tráfego malicioso, enquanto 37% correspondem a tráfego legítimo.

Apesar desse desequilíbrio, optou-se por não aplicar técnicas de *undersampling*, a fim de evitar a perda de amostras representativas de ataques variados, que poderiam comprometer a capacidade do modelo em reconhecer padrões específicos. Da mesma forma, o uso de técnicas de *oversampling*, como a duplicação de instâncias benignas, foi descartado para não introduzir viés artificial e inflar estatísticas de desempenho.

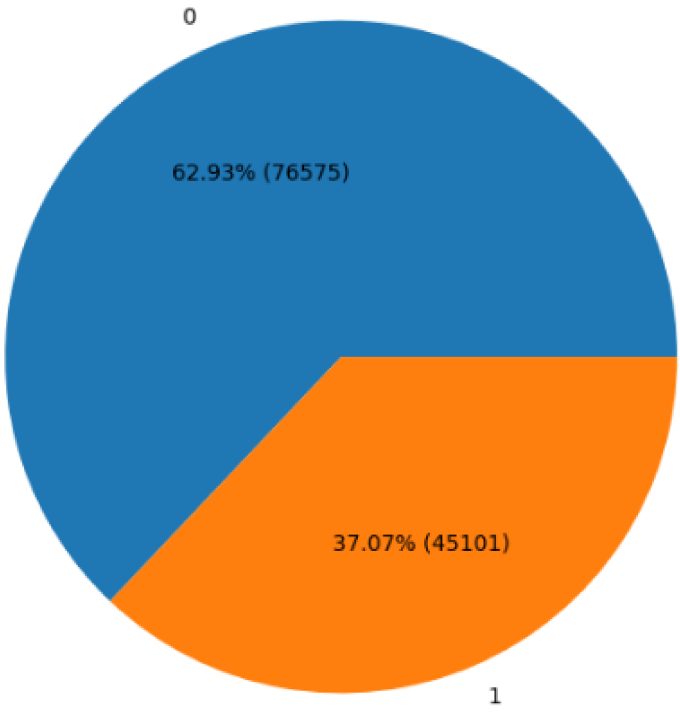

Figura 5 – Proporção das classes binárias

Em seguida, foi realizada a divisão dos dados em conjuntos de treino e teste por meio da técnica de *holdout*, utilizando 70% das amostras para o treinamento e 30% para a avaliação final. Essa separação gerou um total de 85.173 instâncias destinadas ao treinamento dos modelos e 36.503 para o conjunto de teste, preservando a proporção original entre as classes.

5.3.1 Curvas de Aprendizado

Cada modelo foi treinado com subconjuntos progressivamente maiores do total de amostras de treino, mantendo-se fixa a partição de validação. O objetivo foi observar se os modelos apresentavam sinais de sobreajuste (*overfitting*), subajuste (*underfitting*) ou se a performance estabilizava, indicando um bom ponto de generalização.

As Figuras 6 a 10 apresentam as curvas de aprendizado para os modelos Random Forest, XGBoost, C4.5 (com critério de entropia), Decision Tree e MLP, respectivamente. Em cada gráfico, a linha azul representa o desempenho no conjunto de treinamento, enquanto a linha laranja mostra a performance sobre o conjunto de validação. A métrica utilizada foi o *F1-Score*, considerando sua adequação para cenários com desbalanceamento entre classes.

O **Random Forest** (Figura 6) demonstrou desempenho elevado e estável tanto

em treino quanto em validação, com diferença mínima entre as curvas, o que indica boa capacidade de generalização e ausência de sobreajuste relevante. Resultado semelhante foi observado no **XGBoost** (Figura 7), embora com uma leve tendência ao overfitting em tamanhos de treino menores, superada à medida que mais dados foram utilizados.

O **C4.5** (Figura 8) e a **Árvore de Decisão tradicional** (Figura 9) apresentaram maior diferença entre as curvas no início do treinamento, com melhora progressiva à medida que o tamanho do conjunto de treino aumentava. No entanto, uma leve separação persistiu, sugerindo que ambos os modelos são mais propensos ao sobreajuste, especialmente com conjuntos de dados menores.

Por fim, a curva da **Rede Neural MLP** (Figura 10) indicou um comportamento instável. A oscilação nos valores do F1-Score, especialmente no conjunto de validação, pode ser atribuída à sensibilidade da arquitetura utilizada à quantidade de dados, bem como à possibilidade de subajuste causado por uma configuração subótima de hiperparâmetros ou à falta de regularização adequada. Esse comportamento ressalta a necessidade de um ajuste mais refinado quando se utilizam redes neurais em conjuntos com alta dimensionalidade e desbalanceamento.

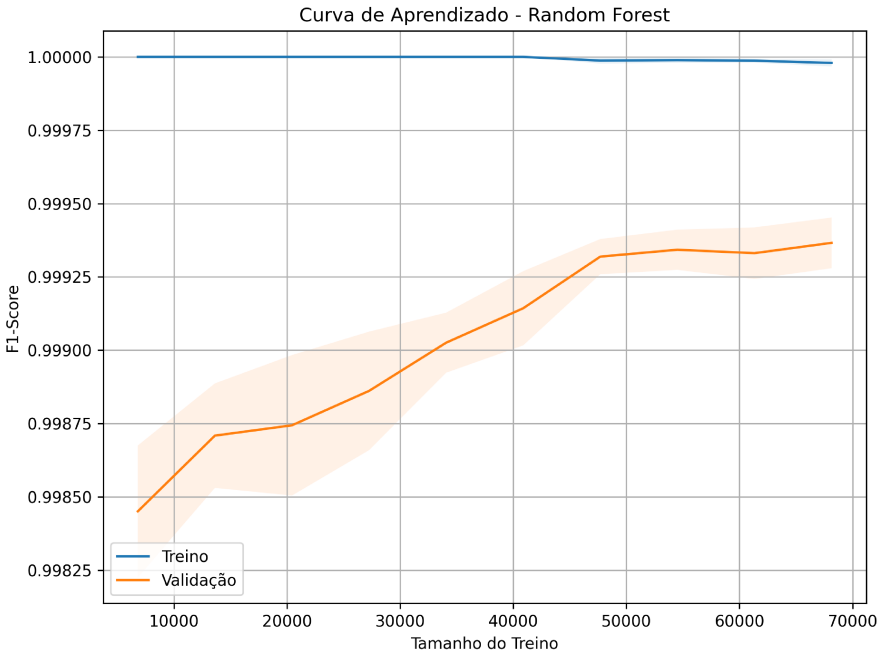

Figura 6 – Curva de aprendizado - Random Forest

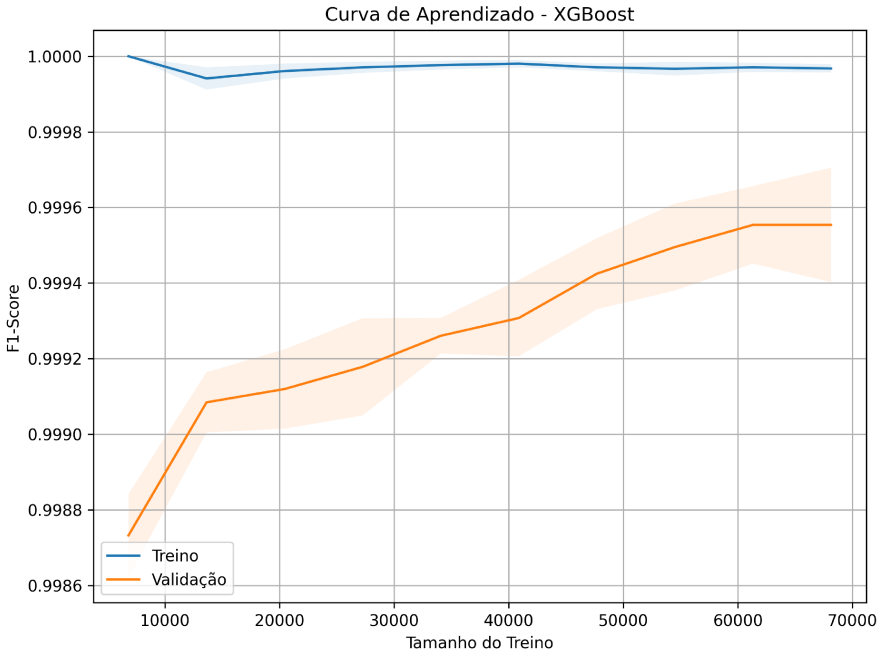

Figura 7 – Curva de aprendizado - XGBoost

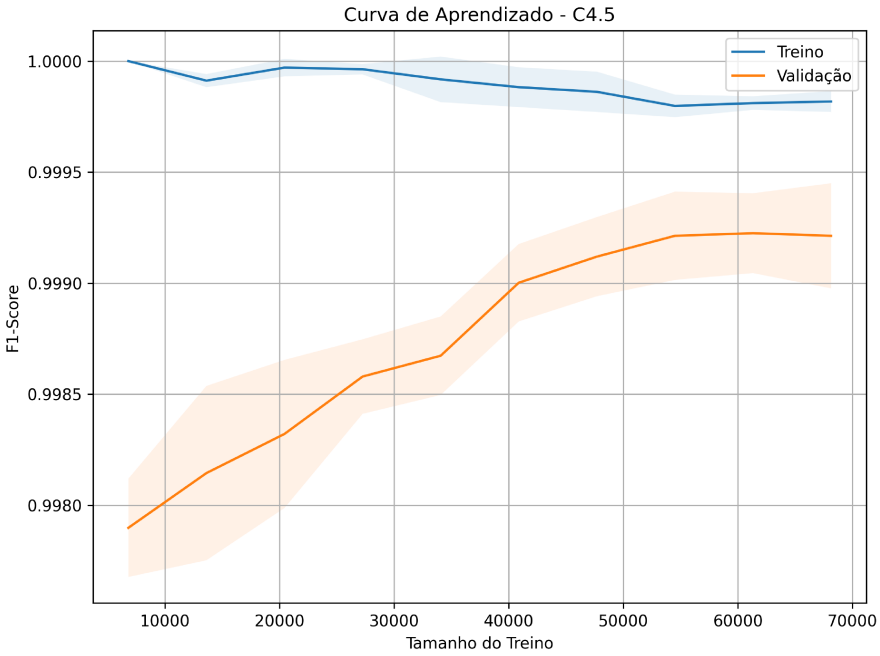

Figura 8 – Curva de aprendizado - C4.5 (Entropia)

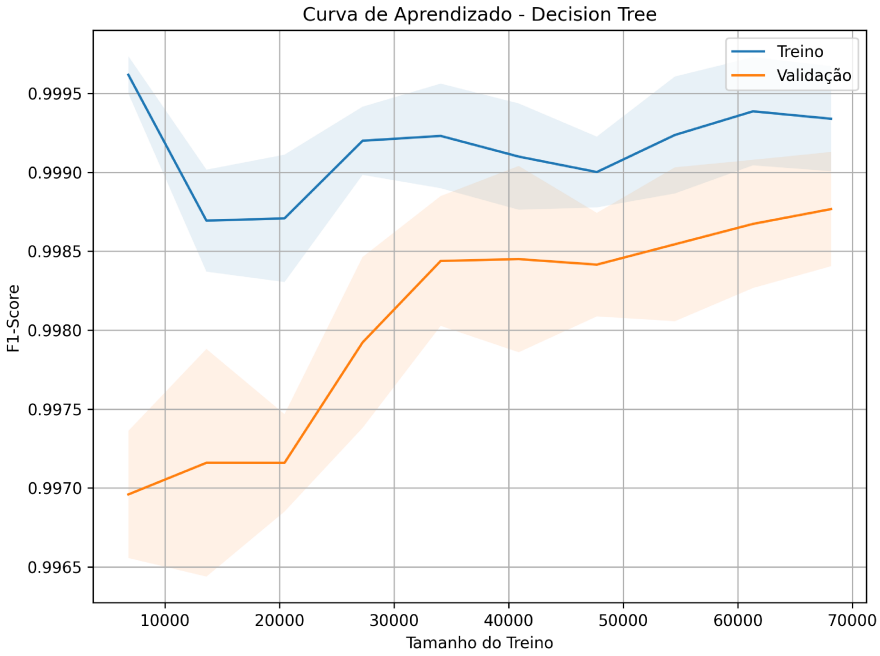

Figura 9 – Curva de aprendizado - Decision Tree

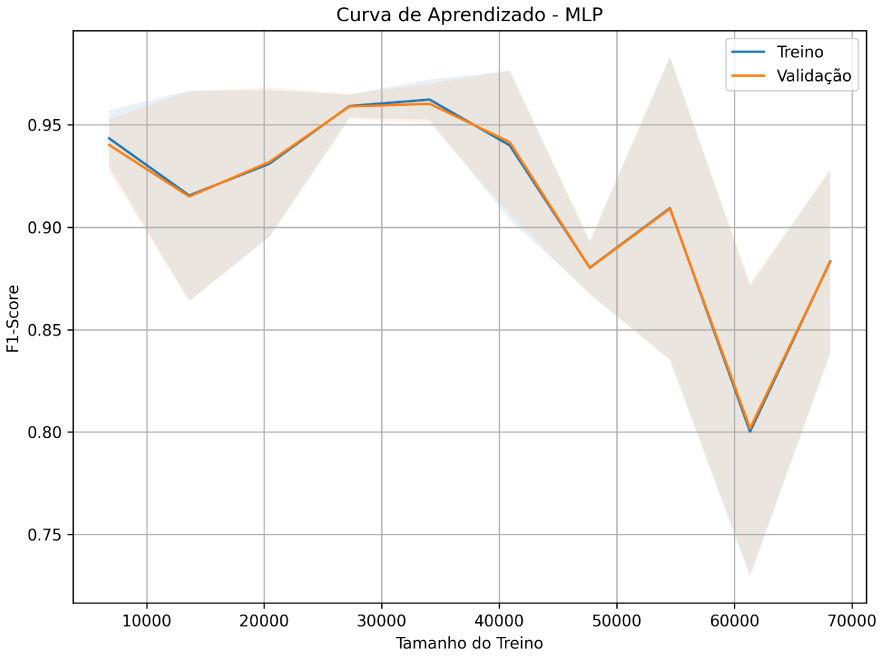

Figura 10 – Curva de aprendizado - MLPClassifier

5.3.2 Comparação dos Erros Residuais

A Figura 11 apresenta a comparação dos erros residuais dos cinco modelos avaliados, utilizando um histograma com escala logarítmica. O eixo horizontal representa os valores dos resíduos: `0` indica uma predição correta, enquanto `-1` e `+1` representam predições incorretas (falso negativo e falso positivo, respectivamente).

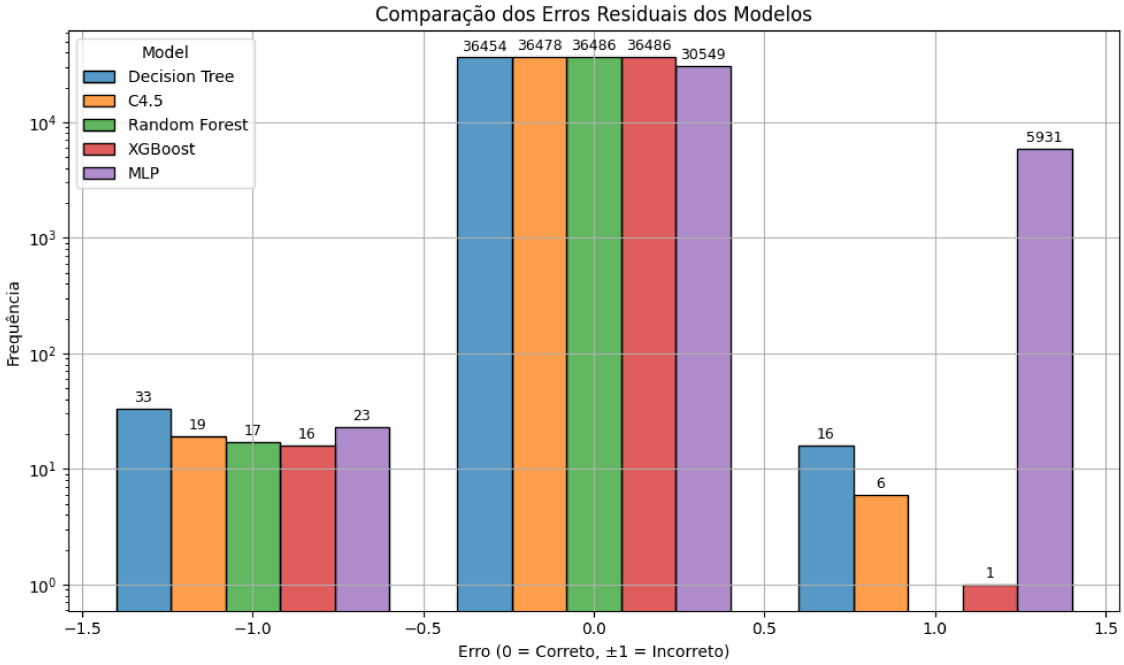

Figura 11 – Comparação dos erros residuais dos modelos (escala logarítmica)

Os modelos baseados em árvores (Decision Tree, C4.5, Random Forest e XGBoost) apresentaram desempenho bastante consistente, com mais de 36.400 predições corretas cada e número reduzido de erros. O **Random Forest** se destacou por não apresentar **nenhum falso positivo** e apenas 17 falsos negativos. Já o **XGBoost** também teve desempenho de destaque, com 16 falsos negativos e um único falso positivo. Assim, os dois modelos apresentam resultados praticamente equivalentes, com variações mínimas no tipo de erro cometido. A escolha entre eles pode ser orientada pelo contexto: o Random Forest é preferível quando se busca minimizar falsos positivos, enquanto o XGBoost pode ser ligeiramente mais eficaz na detecção de ataques reais.

Por outro lado, o modelo **MLP** apresentou o pior desempenho entre os avaliados, com 5.954 predições incorretas, das quais a maior parte foi de falsos positivos. Esse padrão reforça os achados anteriores de menor estabilidade e robustez desse modelo, observados também nas curvas de aprendizado e nas métricas de validação cruzada.

5.3.3 Validação Cruzada (k-fold)

Para avaliar a estabilidade e capacidade de generalização dos modelos, foi realizada validação cruzada com $k = 5$. A Tabela 4 resume os resultados médios do F1-Score e o desvio padrão observado entre as iterações.

Tabela 4 – F1-Score médio na validação cruzada (k=5)

Modelo	**F1 Médio**	**Desvio Padrão**
XGBoost	0.9993	0.0006
Random Forest	0.9990	0.0008
C4.5	0.9985	0.0010
Decision Tree	0.9980	0.0015
MLP	0.8727	0.1042

Os resultados reforçam a consistência dos modelos baseados em árvores, que apresentaram baixíssima variação nas dobras de validação. O modelo MLP, além de apresentar menor desempenho, demonstrou alta instabilidade (desvio padrão de 0,1042). Esses resultados sugerem que o MLP é menos confiável para ambientes de produção onde a estabilidade é crucial.

5.4 Avaliação dos Modelos

A Tabela 5 apresenta os resultados obtidos na partição de teste, incluindo métricas de *Accuracy*, *Precision*, *Recall*, *F1-Score* e *AUC*. O F1-score foi priorizado como métrica principal devido ao desbalanceamento do dataset (63% malicioso, 37% benigno), que pode tornar a acurácia enganadora.

Tabela 5 – Métricas de avaliação dos modelos no conjunto de teste (holdout 30%)

Modelo	**Accuracy**	**Precision**	**Recall**	**F1-Score**	**AUC**
Random Forest	0.9995	0.9995	0.9995	0.9995	0.99997
XGBoost	0.9995	0.9995	0.9995	0.9995	0.99999
C4.5	0.9993	0.9993	0.9993	0.9993	0.99952
Decision Tree	0.9987	0.9987	0.9987	0.9987	0.99948
MLP	0.8369	0.8696	0.8369	0.8234	0.84431

Os modelos baseados em árvores obtiveram desempenho excepcional, com destaque para o *XGBoost*, que apresentou o maior valor de AUC (0,99999). A rede neural MLP, por outro lado, demonstrou desempenho significativamente inferior, possivelmente devido à falta de *tuning* extensivo de hiperparâmetros e à sensibilidade ao desbalanceamento.

Para detalhar o desempenho do melhor modelo, a Figura 12 apresenta a matriz de confusão do XGBoost no conjunto de teste. A matriz mostra 22.957 instâncias maliciosas (classe 0) corretamente classificadas e 13.529 instâncias benignas (classe 1) corretamente identificadas, totalizando apenas 16 falsos positivos (tráfego malicioso classificado como benigno) e 1 falso negativo (tráfego benigno classificado como malicioso). No contexto de detecção de DDoS, os 16 falsos positivos indicam que 16 ataques não foram detectados, o

que, embora pequeno em relação ao total de 36.503 instâncias, é um ponto de atenção, pois representa potenciais ameaças não identificadas. Por outro lado, o único falso negativo significa que apenas 1 instância de tráfego benigno foi classificada como ataque, gerando um alarme falso, mas com impacto mínimo. Esses resultados reforçam a alta capacidade do XGBoost em detectar ataques DDoS com precisão, mantendo um equilíbrio entre minimizar ataques não detectados e alarmes falsos.

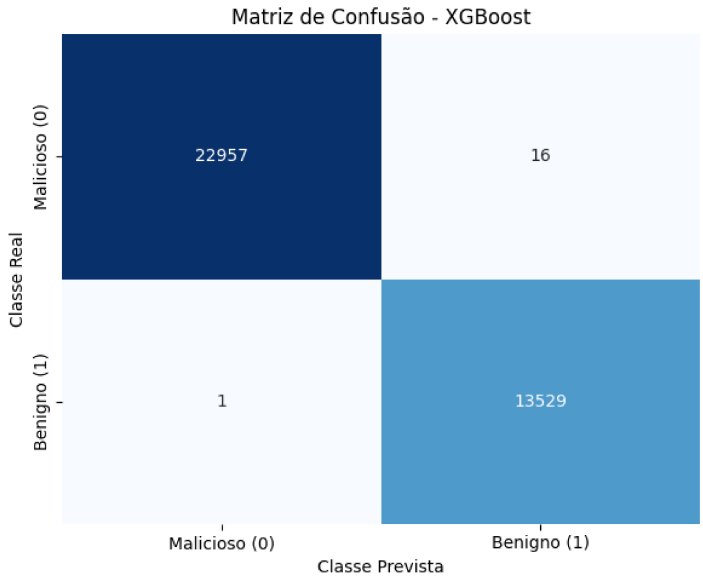

Figura 12 – Matriz de confusão do XGBoost no conjunto de teste

5.5 Análise de Explicabilidade com SHAP

Para complementar a avaliação dos modelos, foi realizada uma análise de explicabilidade utilizando o framework SHAP (SHapley Additive exPlanations), com o objetivo de identificar as *features* mais influentes, avaliar a complexidade dos modelos e selecionar o modelo mais eficiente em termos de simplicidade e interpretabilidade.

Os valores SHAP foram calculados para uma amostra de 100 instâncias do conjunto de treinamento, focando na classe 0 (malicioso). Foram analisadas as seguintes métricas principais:

- Variância média dos SHAP values, como indicador de complexidade do modelo;
- Número de *features* com SHAP médio absoluto superior a 0,01, indicando a dependência do modelo em múltiplas *features*;
- Número de *features* que explicam 80% da importância total, refletindo o grau de concentração da explicabilidade.

A Tabela 6 sumariza os resultados. O Random Forest apresentou a menor variância média dos SHAP values (0,000923), indicando baixa complexidade nas predições e maior estabilidade na atribuição de importância às features. No entanto, depende de 15 *features* com SHAP superior a 0,01, e 12 features são necessárias para explicar 80% da importância, sugerindo uma explicabilidade mais distribuída. O XGBoost, por outro lado, exibiu a maior variância média (1,047607), indicando alta complexidade e maior dependência de interações entre features, com 27 features com SHAP superior a 0,01 e 10 features explicando 80% da importância. Decision Tree, C4.5 e MLP apresentaram variâncias médias moderadas (0,002932, 0,003209 e 0,002801, respectivamente), com 12 a 13 features importantes e 8 features explicando 80% da importância, sugerindo um equilíbrio entre complexidade e interpretabilidade.

Tabela 6 – Comparação de complexidade dos modelos com SHAP

Modelo	**Variância Média SHAP**	**Features (SHAP > 0,01)**	**Features 80%**
Decision Tree	0.002932	13	8
C4.5	0.003209	12	8
Random Forest	0.000923	15	12
XGBoost	1.047607	27	10
MLP	0.002801	12	8

Os gráficos de barras a seguir (Figuras 13 a 17) ilustram a importância média absoluta das features para cada modelo, ordenadas em ordem decrescente. Essa análise permite identificar as *features* mais influentes na detecção de ataques DDoS (classe 0), avaliar a consistência entre os modelos e entender como a escolha do modelo impacta a interpretabilidade das predições. As features com maior impacto médio (eixo Y) indicam maior relevância na classificação, enquanto a distribuição da importância reflete o grau de concentração ou dispersão da explicabilidade.

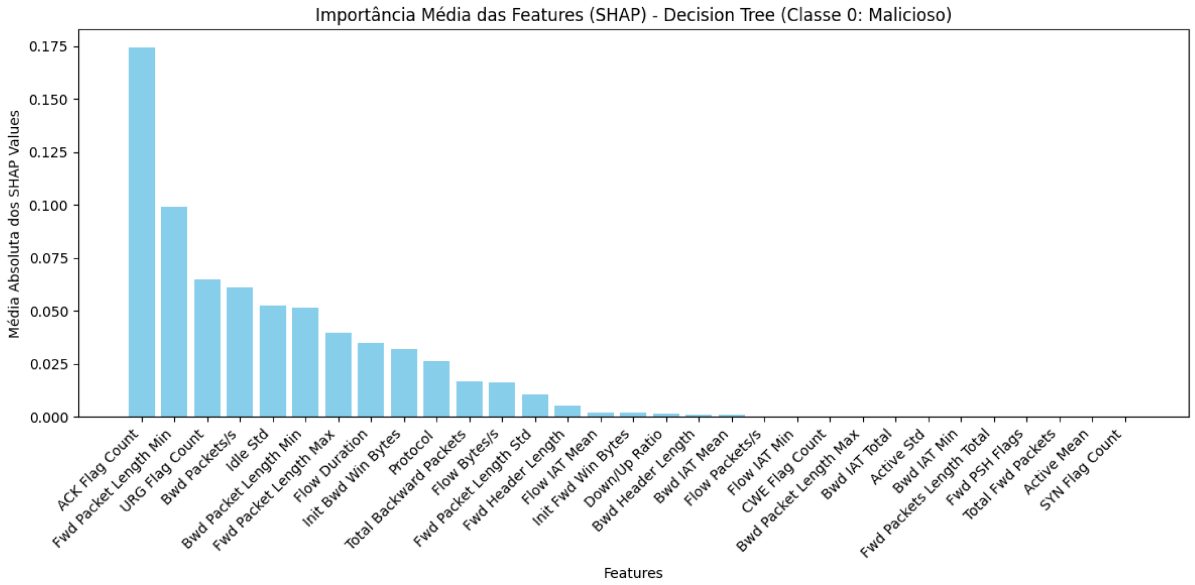

Figura 13 – Importância Média das Features (SHAP) - Decision Tree (Classe 0: Malicioso)

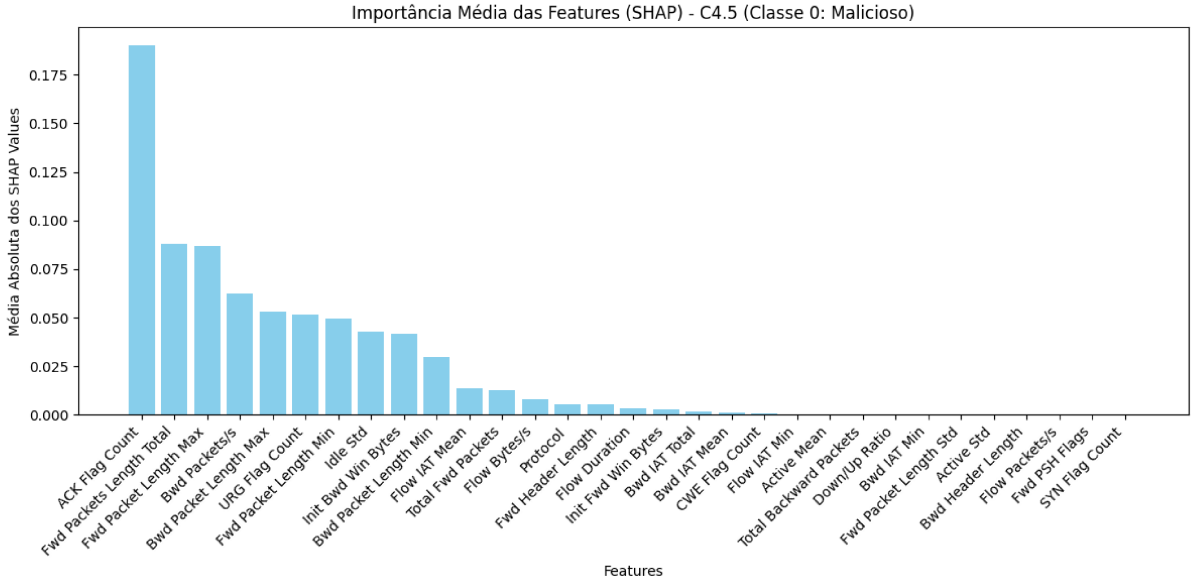

Figura 14 – Importância Média das Features (SHAP) - C4.5 (Classe 0: Malicioso)

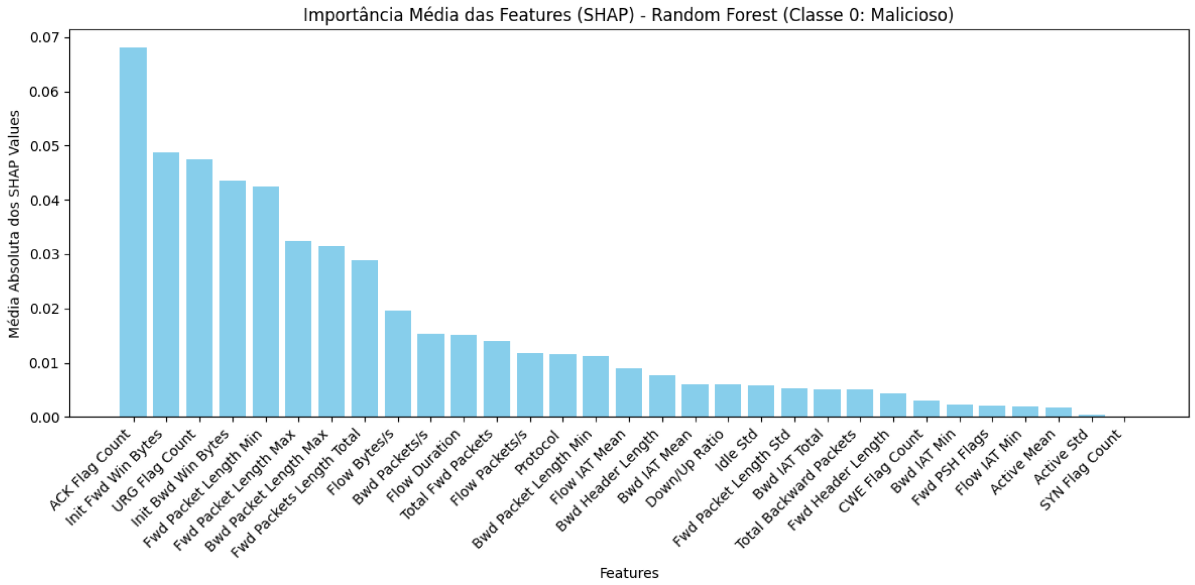

Figura 15 – Importância Média das Features (SHAP) - Random Forest (Classe 0: Malicioso)

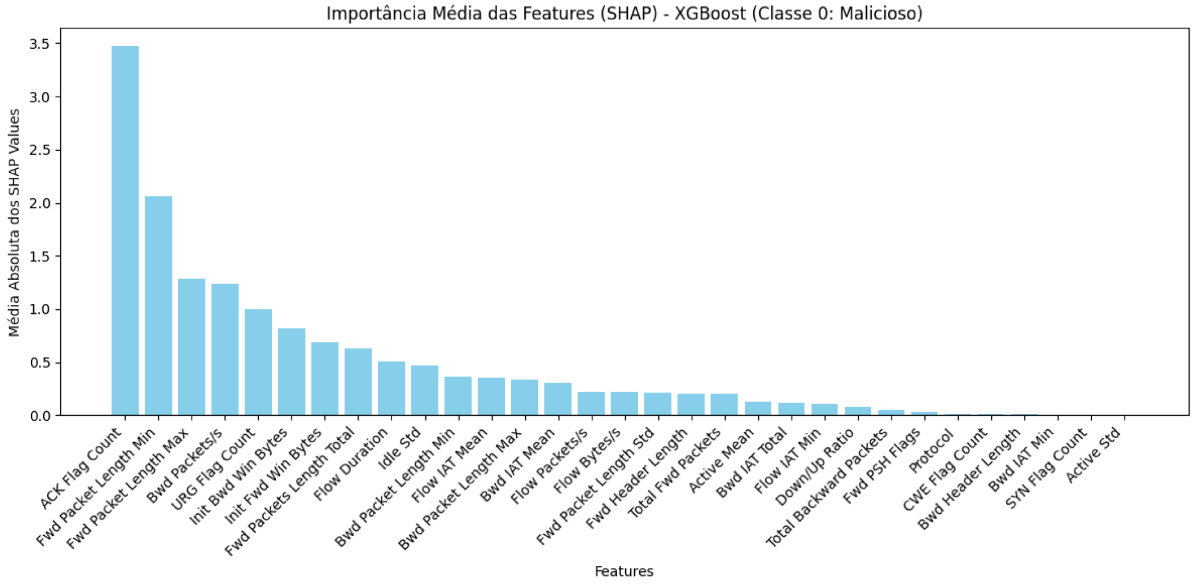

Figura 16 – Importância Média das Features (SHAP) - XGBoost (Classe 0: Malicioso)

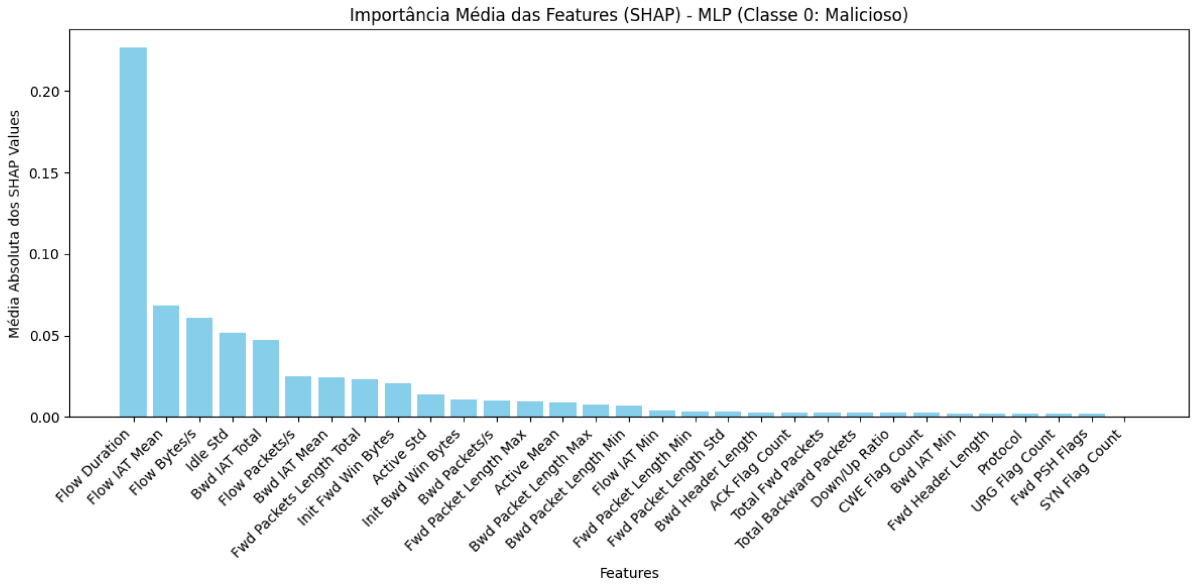

Figura 17 – Importância Média das Features (SHAP) - MLP (Classe 0: Malicioso)

A análise dos gráficos revela uma consistência notável entre os modelos baseados em árvores (Decision Tree, C4.5, Random Forest e XGBoost) em relação às features mais importantes. Features como `ACK Flag Count`, `Fwd Packet Length Min`, `URG Flag Count` e `BWD Packet Length Max/Min` estão entre as mais influentes, com valores médios de SHAP elevados (e.g., `ACK Flag Count` atinge ∼0,175 no C4.5 e ∼3,5 no XGBoost). Essas features estão relacionadas a padrões de tráfego, como flags TCP e tamanhos de pacotes, que são discriminativos em ataques DDoS. O MLP, por outro lado, prioriza features temporais, como `Flow Duration` (SHAP médio ∼0,22) e `Flow IAT Mean`, mas sua instabilidade nas predições (Figura 10) limita sua eficácia, como refletido no F1-score de 0,8234 (Tabela 5).

A distribuição da importância também varia entre os modelos. Decision Tree e C4.5 concentram a explicabilidade nas primeiras 8 features (confirmando a Tabela 6), com uma queda acentuada após as principais, facilitando a interpretação. O Random Forest apresenta uma distribuição mais uniforme, com 12 features contribuindo significativamente, o que reflete sua maior estabilidade (variância média de 0,000923). O XGBoost, embora tenha `ACK Flag Count` como a feature dominante, depende de 27 features com SHAP superior a 0,01, indicando maior complexidade, como esperado pela alta variância média. O MLP, similar ao Decision Tree, concentra a explicabilidade em 8 features, mas sua baixa capacidade de generalização compromete os resultados.

Esses achados reforçam a escolha do Random Forest como o modelo mais eficiente, equilibrando desempenho (F1-score de 0,9995), estabilidade na explicabilidade (baixa variância média) e interpretabilidade prática. As features identificadas, como `ACK Flag Count` e `Fwd Packet Length Min`, estão alinhadas com os padrões de tráfego espe-

rados em ataques DDoS, permitindo não apenas alta precisão na detecção, mas também uma compreensão clara dos fatores que influenciam as predições. No entanto, é importante ressaltar o desempenho do XGBoost, que obteve os melhores resultados em F1-score (0,9995) e AUC (0,99999), conforme Tabela 5. Sua maior complexidade (variância média de 1,047607) e dependência de mais features (27 com SHAP $>$ 0,01) são esperadas, dado que a abordagem deste trabalho optou por agrupar todos os tipos de ataques DDoS em uma classificação binária (ataque ou não), abrangendo diferentes peculiaridades de cada ataque. Essa característica permite ao XGBoost capturar padrões mais diversos e sutis, o que explica seu desempenho ligeiramente superior, mas ao custo de maior dificuldade interpretativa.

6 Conclusão

Este trabalho teve como objetivo investigar a aplicação de algoritmos de aprendizado de máquina na detecção de ataques de negação de serviço distribuídos (DDoS), utilizando o conjunto de dados *CIC-DDoS2019*. Como principal contribuição, destaca-se a otimização do processo de treinamento dos modelos, obtida por meio da redução significativa do número de atributos (features), o que resultou em modelos mais leves e eficientes, com menor consumo de recursos computacionais. A utilização da técnica SHAP foi fundamental nesse processo, permitindo uma seleção racional das variáveis mais relevantes, sem comprometer a acurácia da classificação. Dessa forma, o estudo demonstrou que é possível construir soluções robustas e interpretáveis para a detecção de DDoS - (IDS), mesmo em cenários com restrições de desempenho e recursos.

A recorrente predominância da variável *ACK Flag Count* na explicabilidade dos modelos levanta uma reflexão importante sobre possíveis vieses presentes no conjunto de dados utilizado. Embora, conceitualmente, ataques de negação de serviço do tipo SYN Flood estejam associados majoritariamente ao aumento do flag SYN, a ênfase na contagem de ACK sugere que parte dos ataques no conjunto de dados pode estar associada a tráfego malicioso que explora sessões TCP parcialmente estabelecidas ou simuladas. Esse comportamento pode ser reflexo tanto das características dos vetores de ataque representados no *CIC-DDoS2019* quanto de uma distribuição desbalanceada de certos tipos de tráfego. É possível que a prevalência de padrões específicos, como ataques que manipulam o estado de conexões TCP, tenha induzido os modelos a atribuirem peso elevado a esta *feature*. Esse fenômeno ressalta a necessidade de uma análise crítica sobre a composição e a representatividade do dataset, uma vez que a predominância de determinados atributos pode refletir não apenas características intrínsecas dos ataques DDoS, mas também limitações na diversidade dos cenários simulados. Portanto, embora a variável *ACK Flag Count* se mostre altamente discriminativa dentro deste contexto, é imprescindível considerar que sua relevância pode não generalizar da mesma forma em outros ambientes ou conjuntos de dados, especialmente aqueles que contemplem maior diversidade de vetores de ataque ou tráfego legítimo.

Os resultados confirmaram a superioridade dos modelos baseados em árvores, especialmente os métodos ensemble como *Random Forest* e *XGBoost*, que alcançaram métricas superiores a 99,9% em *F1-score* e *AUC*. Tais números indicam não apenas acurácia, mas também alta capacidade de detecção sem comprometer a taxa de falsos positivos. Em contrapartida, o modelo de rede neural *MLP* apresentou desempenho inferior e maior instabilidade, sinalizando limitações frente à redução do número de *features*.

Também foi resultado deste trabalho compreender os modelos por meio da aplicação da ferramenta SHAP. A análise de explicabilidade permitiu entender os fatores que mais influenciaram as decisões dos classificadores e serviu como critério adicional para avaliar a complexidade de cada algoritmo. Foi possível observar, por exemplo, que modelos como *Decision Tree* e *C4.5* atingiram boa performance utilizando um subconjunto reduzido de atributos relevantes.

Destarte, a combinação entre desempenho preditivo e transparência, viabilizada por técnicas como SHAP, demonstra que soluções inteligentes podem não apenas identificar tráfego malicioso com alta precisão, mas também fornecer justificativas claras para suas decisões. Essa capacidade é essencial para a adoção desses sistemas em ambientes reais, nos quais a confiabilidade, a auditabilidade e a explicação dos resultados são requisitos fundamentais para sua aceitação e aplicação prática.

Trabalhos Futuros

Como continuidade deste estudo, destacam-se as seguintes possibilidades de evolução:

- **Exploração de métodos híbridos**: A combinação de modelos de aprendizado de máquina com técnicas estatísticas (como análise de séries temporais ou detecção baseada em limiares) pode resultar em soluções mais robustas, especialmente para detecção de ataques em tempo real.
- **Implementação em ambiente real**: Uma futura linha de investigação envolve a integração dos modelos desenvolvidos com sistemas de monitoramento de rede reais, a fim de validar sua eficácia em cenários produtivos, com dados não vistos e sob diferentes cargas de tráfego.
- **Detecção contínua e adaptativa**: Incorporar aprendizado online ou incremental, permitindo que os modelos se atualizem continuamente conforme novas ameaças surgem, pode ampliar a longevidade e a eficácia do sistema de detecção.
- **Análise de viés no conjunto de dados e generalização das features**: Investigar em profundidade a influência de variáveis como `ACK Flag Count` na decisão dos modelos, avaliando se essa predominância decorre de características intrínsecas aos ataques DDoS ou de possíveis vieses presentes no conjunto de dados utilizado. Esse estudo pode envolver a utilização de outros datasets, a geração de tráfego sintético controlado ou a análise cruzada com diferentes cenários de rede, de forma a validar se as features identificadas como mais relevantes possuem efetiva capacidade de generalização ou se refletem particularidades específicas do dataset *CIC-DDoS2019*.

Por fim, espera-se que este trabalho contribua para o avanço da pesquisa em detecção de ameaças cibernéticas e incentive o uso de soluções interpretáveis e acessíveis, capazes de fortalecer a cibersegurança em diferentes contextos tecnológicos.

Referências

A10 Networks. *The State of DDoS Weapons*. [S.l.], 2020. 1–2 p. Disponível em: <https://www.a10networks.com/wp-content/uploads/A10-EB-The-State-of-DDoS-Weapons-Report.pdf>. Citado na página 17.

AKAMAI. A year in review — a look at 2023's cyber trends and what's to come. 2023. Disponível em: <https://www.akamai.com/our-thinking/the-state-of-the-internet>. Citado na página 12.

BARROS, P. Aprendizagem de máquina: Supervisionada ou não supervisionada? **Medium**, 2016. Acesso em: 20/10/2023. Disponível em: <https://medium.com/opensanca/aprendizagem-de-maquina-supervisionada-ou-n~ao-supervisionada-7d01f78cd80a>. Citado na página 18.

Cloudflare. **What is a Distributed Denial-of-Service (DDoS) Attack?** 2023. <https://www.cloudflare.com/learning/ddos/what-is-a-ddos-attack/>. Citado na página 16.

CLOUDFLARE. **Record-breaking 5.6 Tbps DDoS attack and global DDoS trends for 2024 Q4**. 2025. Disponível em: <https://blog.cloudflare.com/ddos-threat-report-for-2024-q4/>. Citado na página 16.

D'HOOGE, L.; VERKERKEN, M.; VOLCKAERT, B.; WAUTERS, T.; TURCK, F. D. Establishing the contaminating effect of metadata feature inclusion in machine-learned network intrusion detection models. In: CAVALLARO, L.; GRUSS, D.; PELLEGRINO, G.; GIACINTO, G. (Ed.). **Detection of Intrusions and Malware, and Vulnerability Assessment**. Cham: Springer International Publishing, 2022. p. 23–41. ISBN 978-3-031-09484-2. Citado na página 34.

DOULIGERIS, C.; MITROKOTSA, A. Ddos attacks and defense mechanisms: classification and state-of-the-art. **Computer Networks**, Elsevier, v. 44, n. 5, p. 643–666, 2004. ISSN 1389-1286. Citado na página 17.

FACELI, K.; LORENA, A. C.; GAMA, J.; CARVALHO, A. C. P. L. F. Inteligência artificial: uma abordagem de aprendizado de máquina. In: **Aprendizagem de Máquina: Supervisão ou Não Supervisão?** [S.l.]: P. Barros, 2011. Citado 2 vezes nas páginas 18 e 19.

FAWCETT, T. An introduction to roc analysis. **Pattern Recognition Letters**, Elsevier, v. 27, n. 8, p. 861–874, 2006. Citado na página 20.

G2. **35 DDoS Attack Statistics that Explain Its Rise in 2024**. 2024. Acesso em: [data de acesso]. Disponível em: <https://learn.g2.com/ddos-attack-statistics>. Citado na página 12.

GCORE. **DDoS Attack Trends for Q1–Q2 2024: Insights from Gcore Radar Report**. 2024. Acesso em: [data de acesso]. Disponível em: <https://gcore.com/blog/radar-q1-q2-2024-insights/>. Citado 2 vezes nas páginas 12 e 16.

HAN, D.; LI, H.; FU, X.; ZHOU, S. Traffic feature selection and distributed denial of service attack detection in software-defined networks based on machine learning. **Sensors**, v. 24, n. 13, 2024. ISSN 1424-8220. Disponível em: <https://www.mdpi.com/1424-8220/24/13/4344>. Citado na página 37.

HASSAN, A. I.; REHEEM, E. A. E.; GUIRGUIS, S. K. An entropy and machine learning based approach for ddos attacks detection in software defined networks. **Scientific Reports**, v. 14, p. 18159, 2024. Disponível em: <https://www.nature.com/articles/s41598-024-67984-w>. Citado 2 vezes nas páginas 23 e 27.

KALUTHARAGE, C. S.; LIU, X.; CHRYSOULAS, C.; PITROPAKIS, N. The role of explainable ai in network security: A case study on ddos detection. **MDPI**, 2023. Citado na página 21.

KAUR, A.; KRISHNA, C. R.; PATIL, N. V. A comprehensive review on software-defined networking (sdn) and ddos attacks: Ecosystem, taxonomy, traffic engineering, challenges and research directions. **Computer Science Review**, Elsevier, v. 55, p. 100692, 2025. Citado na página 12.

LABS, F. **2024 DDoS Attack Trends**. 2024. Disponível em: <https://www.f5.com/labs/articles/threat-intelligence/2024-ddos-attack-trends>. Citado na página 16.

MACHADO, V. P. **INTELIGÊNCIA ARTIFICIAL**. 2011. Apostila, 119 páginas. Citado 2 vezes nas páginas 18 e 19.

MAHMOOD, A. A. M. Cybersecurity defence mechanism against ddos attack with explainability. **Mesopotamian Journal of CyberSecurity**, v. 4, n. 3, p. 278–290, Dec. 2024. Disponível em: <https://journals.mesopotamian.press/index.php/CyberSecurity/article/view/678>. Citado na página 23.

NETSCOUT. Ddos threat intelligence report. 2023. Disponível em: <https://www.netscout.com/threatreport>. Citado na página 12.

OSTERWEIL, E.; STAVROU, A.; ZHANG, L. 20 years of ddos: A call to action. **arXiv preprint arXiv:1904.02739**, 2019. Disponível em: <https://arxiv.org/abs/1904.02739>. Citado na página 15.

OTTIS, R. Analysis of the 2007 cyber attacks against estonia from the information warfare perspective. In: ACADEMIC PUBLISHING LIMITED READING, MA. **Proceedings of the 7th European Conference on Information Warfare**. [S.l.], 2008. p. 163. Citado na página 17.

PALMA, B. **ML-SHAP**. 2025. Kaggle. Acesso em: 29/04/2025. Disponível em: <https://www.kaggle.com/code/brenopalma/ml-shap>. Citado na página 32.

_____. **ML-SHAP**. 2025. Disponível em: <https://github.com/BrenoPalma/ML-SHAP>. Citado na página 32.

RAFIQUE, S. H.; ABDALLAH, A.; MUSA, N. S.; MURUGAN, T. Machine learning and deep learning techniques for internet of things network anomaly detection—current research trends. **Sensors**, v. 24, n. 6, 2024. ISSN 1424-8220. Disponível em: <https://www.mdpi.com/1424-8220/24/6/1968>. Citado na página 19.

SAHOSH, Z. H.; FAHEEM, A.; TUBA, M. B.; AHMED, M. I.; TASNIM, S. A. A comparative review on ddos attack detection using machine learning techniques. **Malaysian Journal of Science and Advanced Technology**, p. 75–83, 2024. Citado na página 16.

SANTOS, R.; SOUZA, D.; SANTO, W.; RIBEIRO, A.; MORENO, E. Machine learning algorithms to detect ddos attacks in sdn. **Concurrency and Computation: Practice and Experience**, John Wiley & Sons, Ltd., 2019. Citado na página 21.

SEIFOUSADATI, A. et al. A machine learning approach for ddos detection on iot devices. 2021. Citado 2 vezes nas páginas 25 e 27.

SHARAFALDIN, I.; LASHKARI, A. H.; HAKAK, S.; GHORBANI, A. A. **CIC-DDoS2019**. Kaggle, 2022. Disponível em: <https://www.kaggle.com/dsv/4059918>. Citado 3 vezes nas páginas 28, 33 e 34.

SINGH, J.; BEHAL, S. Detection and mitigation of ddos attacks in sdn: A comprehensive review, research challenges and future directions. **Computer Science Review**, Elsevier, 2020. Citado na página 22.

SOKOLOVA, M.; LAPALME, G. A systematic analysis of performance measures for classification tasks. **Information Processing and Management**, v. 45, n. 4, p. 427–437, 2009. ISSN 0306-4573. Citado na página 19.

STORMWALL. **DDoS in 2024: Detailed Statistics**. 2025. Disponível em: <https://stormwall.network/resources/blog/ddos-attack-statistics-2024#:~:text=In%20total%2C%20StormWall%20systems%20prevented,attack%20intensity%20grew%20by%2053%25.> Citado na página 16.

VILLAçA, J. **Os 10 maiores ataques cibernéticos da História**. 2018. <https://medium.com/ganeshicmc/os-10-maiores-ataques-cibernÃĺticos-da-histÃşria-9803db52462a>. Acesso em: 26/04/2023. Citado na página 16.

WEI, Y.; JANG-JACCARD, J.; SABRINA, F.; CAMTEPE, S. et al. AE-MLP: A hybrid deep learning approach for DDoS detection and classification. **IEEE Access**, IEEE, v. 9, p. 1–1, 2021. License: CC BY-NC-ND 4.0. Citado 2 vezes nas páginas 24 e 27.

ZARGAR, S. T.; JOSHI, J.; TIPPER, D. A survey of defense mechanisms against distributed denial of service (ddos) flooding attacks. **IEEE Communications Surveys and Tutorials**, v. 15, n. 4, p. 2046–2069, 2013. ISSN 1553-877X. Citado 2 vezes nas páginas 15 e 16.

ZEIFMAN, I.; BEKERMAN, D.; HERZBERG, B. **Breaking Down Mirai: An IoT DDoS Botnet Analysis**. 2016. Incapsula.Com, p. 1–11, <https://www.incapsula.com/blog/malware-analysis-mirai-ddos-botnet.html>. Citado 2 vezes nas páginas 16 e 17.

ZHAO, Z.; LIU, Z.; CHEN, H.; ZHANG, F.; SONG, Z.; LI, Z. Effective ddos mitigation via ml-driven in-network traffic shaping. **IEEE Transactions on Dependable and Secure Computing**, v. 21, n. 4, p. 4271–4289, 2024. Citado na página 24.