Gerenciamento Adaptativo de Conexões e Classificação de Endereços IP na Borda da Rede usando Proxies Preditivos AIOps

Francisco V. J. Nobre[1], Ramon S. Araujo[1], Davi O. Alves[1], Janaina R. Santos[1], Jéferson C. Nobre[2], Rafael L. Gomes[1]

[1]Universidade Estadual do Ceará (UECE)

{valderlan.nobre, ramon.araujo, dav.oliveira,

janaina.ribeiro}@aluno.uece.br, rafa.lopes@uece.br

[2]Universidade Federal do Rio Grande do Sul (UFRGS)

jcnobre@inf.ufrgs.br

Abstract. *The increasing dynamism of cyber threats highlights the limitations of static blocklist approaches. Furthermore, real-time querying of multiple Cyber Threat Intelligence (CTI) sources introduces latencies incompatible with the network edge. In this context, this paper proposes an Artificial Intelligence for IT Operations (AIOps) architecture that employs machine learning models as low-latency proxies for multiclass IP address risk classification. A multicriteria consensus mechanism is executed in the cloud to unify data from four CTI sources, forming a robust ground truth. Results demonstrate that the system achieves over 99% parity with the cloud consensus, reducing inference time to under 10 milliseconds. Additionally, evasion tests validate the AIOps cycle, demonstrating the autonomous recovery of predictive efficacy (from 79.9% to 97.5%) in the presence of concept drift.*

Resumo. *A crescente dinamicidade das ameaças cibernéticas evidencia as limitações de abordagens estáticas baseadas em listas de bloqueio. Além disso, a consulta em tempo real a múltiplas fontes de Cyber Threat Intelligence (CTI) introduz latências incompatíveis com a borda da rede. Diante desse cenário, este trabalho propõe uma arquitetura baseada em Artificial Intelligence for IT Operations (AIOps) que utiliza modelos de aprendizado de máquina como proxies de baixa latência para a classificação multiclasse de risco de endereços IP. Um mecanismo de consenso multicritério é executado na nuvem para unificar dados de quatro fontes de CTI, formando uma ground truth robusta. Os resultados demonstram que o sistema alcança mais de 99% de paridade com o consenso da nuvem, reduzindo o tempo de inferência para menos de 10 milissegundos. Adicionalmente, testes de evasão validam o ciclo AIOps, demonstrando a recuperação autônoma da eficácia preditiva (de 79,9% para 97,5%) frente a fenômenos de concept drift.*

1. Introdução

A crescente complexidade e dinamicidade das infraestruturas de rede modernas, especialmente em cenários operacionais de gerenciamento de redes e serviços, têm impulsionado

a adoção de paradigmas emergentes, tais como Inteligência artificial para operações de TI (*Artificial Intelligence for IT Operations* – AIOps) e Inteligência de Ameaças Cibernéticas (*Cyber Threat Intelligence* – CTI) [Brito et al. 2025, Costa et al. 2024]. AIOps combina técnicas de aprendizado de máquina e análise avançada de dados para automatizar tarefas de observabilidade, diagnóstico e tomada de decisão em infraestruturas computacionais [Potts and Carver 2024]. No domínio da cibersegurança, AIOps possibilita a identificação proativa de anomalias, a análise de incidentes e a orquestração automática de respostas a ameaças em larga escala [Yang et al. 2024, Pimenta et al. 2025, Nobre et al. 2025, Souza et al. 2024]. Similarmente, CTI desempenha um papel fundamental na modernização da segurança de rede, ao fornecer dados estruturados e contextuais sobre *Indicators of Compromise* (IoCs) e o comportamento de agentes maliciosos. No contexto do paradigma AIOps, a CTI atua como uma camada vital de enriquecimento de dados [Pimenta et al. 2024, Gomes et al. 2016].

A integração efetiva entre AIOps e CTI ainda apresenta lacunas relevantes, não apenas quanto à adaptação contínua frente à evolução dos atacantes (*Concept Drift*) [Wagner et al. 2019, Ferreira et al. 2024], mas substancialmente em relação a gargalos operacionais. Abordagens tradicionais requerem a consulta em tempo real a múltiplas fontes públicas de inteligência para formar um consenso sobre um endereço IP. Em ambientes de rede de alto desempenho, como *gateways* e roteadores de borda, aguardar a latência de rede decorrente de requisições a múltiplas *Application Programming Interfaces* (APIs) de terceiros degrada significativamente a Qualidade de Serviço (QoS), limitando a viabilidade de mitigações em tempo real [Portela et al. 2024, Gomes et al. 2013].

Trabalhos recentes têm explorado o uso de aprendizado de máquina para a classificação de endereços IP maliciosos, combinando características de tráfego, métricas de reputação e dados CTI [Usman et al. 2021, Huang et al. 2023]. Embora apresentem alta acurácia, essas abordagens frequentemente assumem cenários onde o tempo de processamento não é o gargalo, dependem de mecanismos complexos e lentos de fusão de dados ou carecem de estratégias arquiteturais que dissociem a análise pesada (nuvem) da decisão em milissegundos (borda) [Gama et al. 2014].

Diante dessas limitações operacionais, este trabalho propõe uma arquitetura baseada em AIOps fundamentada no princípio de *Knowledge Distillation*. Utilizam-se modelos de aprendizado de máquina não como descobridores independentes de ameaças, mas como *proxies* de baixa latência treinados para emular a classificação de conexões IP ditada pela nuvem. Ao invés de realizar consultas web custosas durante o fluxo de tráfego, propõe-se uma arquitetura híbrida: na nuvem, um mecanismo de votação ponderada multicritério funde dados de quatro fontes públicas de CTI para construir de forma offline uma *ground truth* robusta; na borda, modelos de *ensemble learning* previamente treinados sobre esses dados operam como destiladores desse conhecimento, ”imitando”as regras de consenso da nuvem. Isso permite classificar as conexões (*allowlist*, *denylist* ou *suspicious*) diretamente nos equipamentos de rede, substituindo segundos de latência de rede por inferências locais, com monitoramento contínuo para detecção de *concept drift*.

As principais contribuições deste trabalho são: (i) a proposição e avaliação em ambiente de simulação de uma arquitetura AIOps híbrida (nuvem-borda) que utiliza aprendizado de máquina como *proxy* de baixa latência, viabilizando decisões de mitigação multiclasse em tempo quase real; (ii) a construção e disponibilização de um conjunto de

dados enriquecido por um mecanismo de consenso multicritério (reduzindo conflitos e ruídos de fontes públicas isoladas); (iii) a avaliação sistemática da viabilidade operacional de modelos supervisionados e estratégias de *ensemble* frente a restrições de tempo de inferência; e (iv) a integração de um mecanismo de monitoramento de desempenho para adaptação ao *concept drift*.

O restante deste artigo está organizado da seguinte forma: a Seção 2 apresenta os trabalhos relacionados; a Seção 3 descreve a arquitetura proposta; a Seção 4 detalha a metodologia experimental e os dados; a Seção 5 discute os resultados obtidos; e a Seção 6 apresenta as conclusões.

2. Trabalhos Relacionados

A literatura recente sobre detecção e classificação de ameaças em redes apresenta diversas estratégias para lidar com o volume e a dinamicidade dos ataques. Para evidenciar o posicionamento científico desta pesquisa, os trabalhos correlatos são analisados sob três eixos metodológicos principais: (i) abordagens baseadas em tráfego e grafos; (ii) abordagens baseadas em fusão de Inteligência de Ameaças Cibernéticas (CTI); e (iii) mecanismos adaptativos operacionais. Uma síntese comparativa é apresentada na Tabela 1.

2.1. Abordagens baseadas em Tráfego e Grafos.

Diversos trabalhos buscam extrair padrões diretamente dos pacotes de rede ou das relações topológicas. Yang e Lim [Yang and Lim 2021] desenvolveram uma técnica de aprendizado profundo para detecção de tráfego *Secure Sockets Layer* (SSL) malicioso, reconstruindo registros a partir de pacotes capturados. Embora eficaz para tráfego cifrado, o método ignora dados estruturados de CTI, limitando a identificação proativa de novas campanhas. Huang et al. [Huang et al. 2023] propuseram o uso de Redes Neurais de Grafos (GNN) para correlacionar serviços (e-mail, web, DNS) e características como sub-redes e geolocalização. Apesar da inovação relacional, o método impõe elevada sobrecarga computacional e requer disponibilidade simultânea de múltiplos protocolos, dificultando sua aplicação em roteadores de borda.

2.2. Abordagens baseadas em Fusão de CTI.

A integração de múltiplas fontes públicas tem sido explorada para aumentar a robustez da classificação. Usman et al. [Usman et al. 2021] propuseram um sistema híbrido que integra aprendizado de máquina, análise dinâmica de *malware* e CTI (VirusTotal, AlienVault). Apesar da alta acurácia na tipificação de ameaças, o sistema apresenta forte acoplamento a ambientes de *sandbox*, cujo custo computacional inviabiliza respostas em tempo real. De forma mais alinhada a cenários operacionais, Siam et al. [Siam et al. 2025] apresentaram o *IP SafeGuard*, que funde CTI adaptativamente usando *XGBoost* sobre características de DNS Passivo e NetFlow. Spyros et al. [Spyros et al. 2025] avançaram ao propor o *ThreatWise AI*, integrando processamento de linguagem natural sobre fontes não estruturadas e armazenando resultados em plataformas MISP. No entanto, ambas as soluções dependem de *pipelines* de extração de dados complexos que geram gargalos de latência no fluxo crítico da rede.

2.3. Mecanismos Adaptativos e Governança.

A evolução temporal das ameaças (*concept drift*) é um desafio amplamente reconhecido em Sistemas de Detecção de Intrusão em Redes (NIDS) adaptativos [Gama et al. 2014],

mas raramente integrado à operacionalização de CTI. Park et al. [Park et al. 2024] analisaram os requisitos de segurança para sistemas de IA autônomos, mas a abordagem restringiu-se ao nível normativo e de auditoria, distanciando-se da implementação de bloqueios automáticos.

2.4. A Lacuna Científica

A análise da literatura revela um compromisso (*trade-off*) recorrente, abordagens que utilizam inteligência rica e multicritério exigem consultas web e *pipelines* lentos (segundos a minutos), enquanto sistemas capazes de atuar na borda da rede dependem de sinais simples ou locais que perdem rapidamente a eficácia frente ao *concept drift*. Nossa proposta preenche essa lacuna ao utilizar algoritmos de *ensemble learning* não como extratores analíticos primários, mas como ***proxies* de baixa latência** que destilam e aproximam o consenso das fontes CTI pesadas da nuvem. Isso transfere a complexidade computacional para o treinamento *offline*, permitindo que a inferência na borda ocorra de forma rápida (abaixo de 10 ms), com métricas integradas para acionar retreinamentos frente à evolução das ameaças.

Tabela 1. Resumo Comparativo dos Trabalhos Relacionados

Trabalho	Metodologia	Integração de CTI	Adequação à Borda
[Usman et al. 2021]	Híbrido (Sandbox + Árvores)	Múltiplas fontes	Não (Alta latência)
[Yang and Lim 2021]	*Deep Learning* (Payload)	Nenhuma	Parcial
[Huang et al. 2023]	Grafos Heterogêneos	Indireta (ASN/Geo)	Não (Custo computacional)
[Siam et al. 2025]	Fusão ML (*XGBoost*)	Múltiplas fontes	Sim
[Spyros et al. 2025]	NLP + Análise de Anomalias	Fontes não-estruturadas	Não (*Pipeline* pesado)
Nossa Proposta	**AIOps: ML como *Proxy***	**Consenso Multicritério**	**Sim (*Ensemble*)**

3. Proposta

Este trabalho propõe uma arquitetura híbrida e distribuída baseada no ciclo AIOps (Observação–Análise–Ação) para o gerenciamento e classificação de endereços IP maliciosos. A estrutura é dividida em camadas de Computação em Nuvem e Computação em Borda, conforme ilustrado na Figura 1. O objetivo central da arquitetura é resolver o compromisso (*trade-off*) entre inteligência multicritério e latência de rede, enquanto a Nuvem realiza o processamento analítico intensivo e assíncrono, a Borda utiliza modelos de aprendizado de máquina como *proxies* de curtíssima latência para mitigações em tempo quase real.

O fluxo operacional do sistema é orquestrado por meio de três módulos principais, distribuídos entre os ambientes lógicos:

- **Verificação de Reputação (Nuvem):** Responsável por consultar assincronamente nas APIs de bases públicas de ameaças (AbuseIPDB, VirusTotal, APIVoid e Pulsedive) para endereços de IP históricos e recém-descobertos. Os atributos brutos (ex: índices de reputação, contagem de detecções e relatórios geolocalizados) já disponíveis nestas plataformas são coletados e submetidos a um mecanismo de Votação Ponderada Multicritério (detalhado na Seção 4). Este processo gera uma *ground truth* de alta confiabilidade, evitando a dependência de avaliações isoladas.

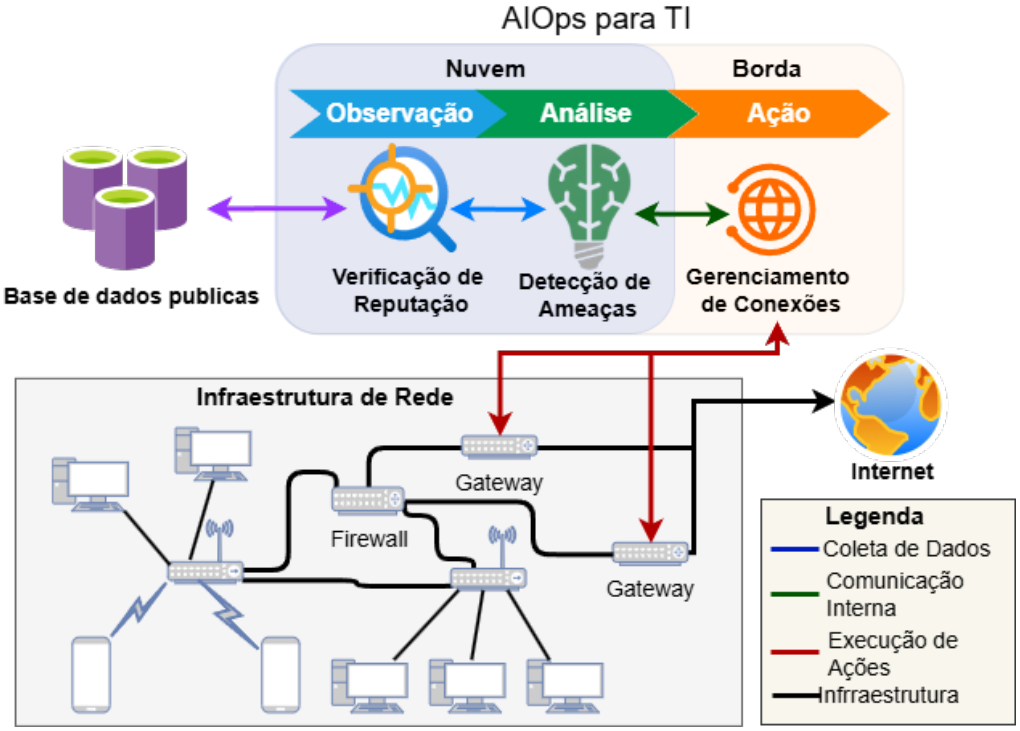

Figura 1. Visão Geral da Arquitetura AIOps Nuvem-Borda.

- **Detecção de Ameaças (Nuvem):** Atua como o motor de Inteligência Artificial do sistema. Utilizando a base rotulada pela etapa de consenso, este módulo treina *offline* modelos supervisionados de *ensemble*. O propósito desta etapa é realizara síntese preditiva do mecanismo de votação, o algoritmo aprende a replicar matematicamente as regras de maior custo computacional e latência de consenso baseadas em internet, gerando um modelo preditivo compacto.
- **Gerenciamento de Conexões (Borda):** Localizado próximo à infraestrutura física (em roteadores ou *gateways* de rede), este componente embarca o modelo de ML previamente treinado na nuvem. Ao interceptar conexões, o módulo extrai metadados locais ou acessa atributos em *cache* e executa a inferência preditiva em milissegundos. Baseado na classificação (*allowlist*, *suspicious* ou *denylist*), o *gateway* executa ações imediatas, tais como: (i) liberação do fluxo; (ii) Degradação Temporária de Conexão (*Tarpit*) para reduzir a prioridade de IPs suspeitos; ou (iii) bloqueio sumário com atualização de tabelas de roteamento e *firewall*.

Para garantir resiliência frente à evolução das ameaças, a arquitetura integra detecção de *concept drift*. O módulo de borda reporta telemetria de inferências à nuvem e, caso detecte degradação preditiva (como queda de acurácia ou anomalias na distribuição de classes), o ciclo AIOps reinicia automaticamente: novas CTI são coletadas, o consenso é recalculado e os modelos retreinados. Por fim, os *gateways* recebem os pesos atualizados de forma transparente, adaptando-se sem interromper o tráfego.

4. Experimento

Esta seção descreve o protocolo experimental adotado para avaliar o desempenho dos modelos de classificação de endereços IP. São detalhados o ambiente computacional, os

modelos avaliados, as estratégias de *ensemble* e as métricas utilizadas para a análise dos resultados obtidos. Com o objetivo de garantir a reprodutibilidade científica, o código-fonte da solução proposta, bem como o conjunto de dados utilizado nos experimentos, estão publicamente disponíveis em um repositório online[1].

4.1. Ambiente Computacional

Os experimentos foram conduzidos em uma estação de trabalho equipada com processador Intel Core i9-14900, 64 GB de memória RAM e armazenamento SSD NVMe de 1 TB. O sistema operacional utilizado foi o Ubuntu 24.04 LTS. Todo o desenvolvimento foi realizado em Python 3.13, com gerenciamento de dependências por meio de ambiente virtual isolado.

4.2. Formação, Rotulagem e Balanceamento do Conjunto de Dados

Um conjunto de dados foi construído com o objetivo de treinar o modelo proposto a partir de informações provenientes dos bancos de dados públicos mencionados. Por meio de um *bot* de coleta automatizado, reuniram-se 10.000 endereços IP, inicialmente descritos por 26 atributos brutos de múltiplos provedores de CTI, incluindo informações geográficas (por exemplo, códigos de país) e marcações temporais (*timestamps*). Após um processo de seleção de características para remover metadados não informativos, 11 atributos relevantes foram mantidos para o treinamento. Dentre os IPs coletados, 5.806 pertencem a *denylists* do AbuseIPDB, enquanto 4.194 correspondem a endereços presentes em *allowlists* mantidas por provedores confiáveis, como Google, Amazon e Microsoft. As características selecionadas para o treinamento dos modelos e seus respectivos domínios de valores são detalhadas a seguir:

- **abuseipdb_confidence_score** (0 a 100): Índice de confiança para a probabilidade de atividade maliciosa, baseado no histórico de denúncias. Valores próximos a 100 indicam forte evidência de comportamento malicioso.
- **abuseipdb_total_reports** (0 a 35.000): Total de denúncias de atividade maliciosa registradas contra o endereço IP. Reflete a frequência com que o endereço é sinalizado como ameaça.
- **abuseipdb_num_distinct_users** (0 a 1.100): Quantidade de usuários distintos que reportaram o IP, fortalecendo a credibilidade da denúncia.
- **apivoid_risk_score** (0 a 100): Estimativa de risco calculada pelo *APIVoid*, que combina a análise de múltiplas *denylists* e métricas de reputação para atribuir uma pontuação de periculosidade ao endereço IP.
- **apivoid_blacklists_detection_rate** (0 a 1): Taxa de detecção que representa a proporção de motores de *denylist* que identificaram o IP como malicioso em relação ao total de motores consultados.
- **risk_recommended_pulsedive**: Classificação de risco consolidada (*critical, high, medium, low, unknown, none*) a partir de múltiplas fontes de análise.
- **virustotal_reputation** (-127 a 565): Pontuação de reputação onde valores negativos indicam má reputação do IP baseada em relatórios de segurança.
- **virustotal_malicious** (0 a 95): Número de mecanismos que consideram o IP como malicioso.

[1] `https://github.com/LarcesUece/AIOps-CTI-Ensemble`

- **virustotal_suspicious** (0 a 95): Número de mecanismos que categorizam o comportamento do IP como suspeito.
- **virustotal_undetected** (0 a 95): Número de mecanismos que não detectaram ameaças.
- **virustotal_harmless** (0 a 95): Número de mecanismos que consideram o endereço IP inofensivo.

Após a coleta inicial, os dados foram submetidos a uma etapa de pré-processamento, que incluiu a normalização de características numéricas via escala *min-max* e a categorização de atributos qualitativos. Na sequência, para garantir a rotulação confiável das amostras, aplicou-se o método de Votação Ponderada Multicritério (detalhado na Seção 4.3). Esse processo classificou cada endereço IP em uma das três classe de risco estabelecidas: *denylist* (4.282 amostras), *allowlist* (4.128 amostras) e *suspicious* (1.590 amostras). O conjunto de dados foi particionado em 80% para treinamento e 20% para teste, garantindo que a avaliação dos modelos fosse realizada sobre dados não vistos durante o aprendizado. Para tratar o desbalanceamento entre as classes, aplicou-se a técnica *Synthetic Minority Over-sampling Technique* (SMOTE) exclusivamente sobre o conjunto de treinamento, promovendo a geração sintética de amostras da classe minoritária. Essa abordagem possibilitou uma distribuição mais equilibrada entre as três classes.

Em situações de indisponibilidade temporária de fontes públicas ou valores ausentes, os atributos são imputados com base na mediana dos endereços IP mais recentes pertencentes à mesma região geográfica. Essa escolha metodológica, em detrimento do uso da média, justifica-se pela robustez da mediana frente a valores discrepantes inerentes ao tráfego anômalo e a ataques cibernéticos. Dessa forma, evita-se a introdução de vieses, preservando a consistência estatística do conjunto de dados e garantindo a robustez operacional do sistema frente a falhas externas.

4.3. Votação Ponderada Multicritério

A formação de uma *ground truth* confiável a partir de fontes públicas de CTI representa um desafio relevante, uma vez que diferentes bases frequentemente apresentam informações incompletas, divergentes ou desatualizadas sobre um mesmo endereço IP [Wagner et al. 2019]. Para mitigar esse desafio, este trabalho emprega um mecanismo de Votação Ponderada Multicritério como etapa inicial de rotulagem do conjunto de dados, evitando a dependência de um único *score* ou de decisões isoladas de fontes individuais.

A estratégia fundamenta-se no mecanismo de Votação Ponderada Multicritério formalizado previamente em [Nobre et al. 2026], o qual adapta os princípios matemáticos clássicos de *ensemble learning* [Dietterich 2000, Kuncheva 2004], nos quais decisões oriundas de múltiplos especialistas são combinadas de forma a aumentar a robustez e a confiabilidade do resultado final. Cada fonte de CTI contribui com um voto ponderado de acordo com três critérios complementares: (i) seu desempenho histórico individual na discriminação de IPs maliciosos; (ii) o grau de correlação entre suas saídas e o consenso global observado; e (iii) o nível de concordância correta com as demais fontes, refletindo consistência inter-fontes.

Formalmente, para um endereço IP x, a classe atribuída corresponde àquela que maximiza a soma dos pesos das fontes que votam nessa classe, conforme a Equação 1:

$$Class(x) = \arg\max_{c \in \mathcal{C}} \sum_{i \in V_c(x)} w_i \quad (1)$$

onde $\mathcal{C} = \{\textit{allowlist}, \textit{suspicious}, \textit{denylist}\}$, $V_c(x)$ representa o conjunto de fontes que atribuem a classe c ao IP x, e w_i denota o peso associado à fonte i. Os pesos são normalizados e definidos a partir da combinação linear dos critérios mencionados, com coeficientes ajustados empiricamente em um conjunto de validação.

É importante destacar que o mecanismo de Votação Ponderada Multicritério é empregado exclusivamente na etapa de construção e atualização do conjunto de dados, não sendo utilizado diretamente na classificação online das conexões. Sua função é fornecer uma base rotulada mais consistente e robusta, que sustente o treinamento inicial dos modelos supervisionados e possibilite atualizações periódicas frente a fenômenos de *concept drift*, sem impactar o fluxo de decisão em tempo real.

4.4. Simulação de *Concept Drift* e Ameaças à Validade

Para validar os mecanismos de AIOps propostos, modelou-se uma simulação temporal de *concept drift* com isolamento estrito de dados, evitando vazamento temporal. O conjunto de dados foi ordenado cronologicamente e dividido em quatro janelas: T_1 (treinamento passado, 50%), T_2 (teste pré-*drift*, 20%), T_{3A} (início do ataque *Zero-Day*, 15%) e T_{3B} (teste de recuperação pós-retreinamento, 15%). Nas janelas T_{3A} e T_{3B}, aplicou-se uma perturbação agressiva para emular uma campanha massiva de evasão, 70% dos endereços IP reais de ameaças foram manipulados para ofuscar completamente seus rastros nas duas fontes de CTI mais fortes do modelo (AbuseIPDB e VirusTotal), forçando notas nulas de detecção e reputação máxima. O objetivo foi testar a resiliência da arquitetura frente à falha falha severa (ou indisponibilidade crítica) de seus principais provedores de inteligência.

A degradação do desempenho do modelo em T_{3A} foi monitorada por meio da acurácia, sendo definido um limiar mínimo de desempenho $\tau = 0{,}92$ como critério para detecção de perda de eficácia preditiva. Quando a acurácia observada cair abaixo desse limiar, o sistema aciona o mecanismo de retreinamento na nuvem, fundindo os dados recentes e reiniciando o ciclo AIOps. Avaliou-se, portanto, não apenas a queda de desempenho do modelo envelhecido (T_{3A}), mas também a recuperação da eficácia preditiva em tráfego futuro (T_{3B}) após a atualização do modelo.

No que tange às ameaças à validade interna, destaca-se o aparente vazamento de dados (*data leakage*) decorrente da **dependência estrutural entre atributos e rótulos**. Como o *ground truth* deriva analiticamente das próprias métricas de CTI utilizadas como entrada, o sistema diverge da ótica clássica de predição cega. Em vez disso, a arquitetura consolida-se como uma *Knowledge Distillation*. O objetivo inerente do modelo *proxy* na borda é aprender a replicar o comportamento do mecanismo de consenso multicritério. Assim, a alta acurácia observada não indica um viés metodológico, mas atesta o sucesso da técnica em transferir a inteligência centralizada da nuvem para inferências locais em milissegundos.

Como ameaça à validade externa, ressalta-se que o tempo de inferência reportado reflete o custo computacional isolado do modelo, assumindo que as características já se encontram disponíveis em memória no *gateway*. Dessa forma, não são contabilizadas

latências adicionais associadas ao plano de dados, como enfileiramento de pacotes TCP ou limitações específicas de *hardware*.

4.5. Aprendizado de Máquina e Métricas de Avaliação

Ao todo, foram avaliados oito classificadores supervisionados, abrangendo modelos individuais baseados em árvores, distância, margens e redes neurais, além de estratégias de *ensemble learning*. Essa diversidade garante uma comparação abrangente sob diferentes paradigmas de aprendizado, considerando tanto desempenho preditivo quanto viabilidade operacional em cenários de segurança de rede.

A etapa de Aprendizado de Máquina foi conduzida a partir da avaliação de modelos representativos desses diferentes paradigmas, selecionados com base em sua recorrente aplicação em problemas de classificação de dados tabulares e cibersegurança [Costa et al. 2021, Lazar et al. 2021]. Entre os modelos individuais, foram considerados classificadores baseados em árvores, como a *Decision Tree* (DT), utilizada como modelo de referência devido à sua simplicidade e baixa latência, e *Random Forest* (RF), amplamente empregada por sua robustez ao ruído e elevada capacidade de generalização.

Os modelos baseados em margens e distância, especificamente o *Support Vector Machine* (SVM) e o *k-Nearest Neighbors* (KNN), que oferecem perspectivas complementares de decisão, especialmente em cenários com fronteiras de classe não lineares. No grupo de modelos neurais, foram incluídas uma *Neural Network* (NN), capaz de capturar relações não lineares entre os atributos, e uma *Convolutional Neural Network* (CNN) adaptada para vetores unidimensionais de características, explorando padrões locais nos dados de Inteligência de Ameaças.

Este trabalho empregou estratégias de *Ensemble Learning* para maximizar o desempenho preditivo e reduzir as variações nos dados. A abordagem de votação suave (*soft voting*) combinou as probabilidades preditas pelos modelos base, enquanto a técnica de *stacking* utilizou uma RF como meta-classificador, treinada sobre as saídas dos classificadores individuais obtidas por validação cruzada, prevenindo vazamento de informação entre os conjuntos de treinamento e teste.

Os hiperparâmetros de todos os modelos avaliados foram otimizados por meio de *Grid Search* com validação cruzada estratificada, buscando um equilíbrio entre complexidade do modelo e capacidade de generalização. Essa etapa foi essencial para evitar sobreajuste e garantir comparações justas entre as abordagens. Os valores finais dos hiperparâmetros selecionados são apresentados na Tabela 2.

A avaliação experimental considerou métricas adequadas a cenários multiclasse e potencialmente desbalanceados. Foram utilizadas Acurácia e Acurácia Balanceada para mensurar o desempenho global, enquanto o F1-Macro foi adotado para atribuir igual importância às três classes, com especial atenção à classe *suspicious*, fundamental para decisões graduais de mitigação em ambientes operacionais. O *Recall* permitiu analisar a sensibilidade dos modelos por classe, e o MCC foi empregado por refletir a correlação global entre predições e rótulos reais, mesmo em cenários complexos. Adicionalmente, o *Log Loss* foi utilizado para avaliar a calibração probabilística das predições, enquanto o *Overfit Gap* possibilitou quantificar diferenças de desempenho entre os conjuntos de treinamento e teste, fornecendo indícios sobre a robustez e a estabilidade dos modelos.

Tabela 2. Hiperparâmetros Otimizados

Modelo	Hiperparâmetro	Valor	Hiperparâmetro	Valor
Stacking	max_depth (final_estimator)	10	n_estimators (final_estimator)	50
Voting	voting	soft	weights	[2, 1, 1]
CNN	conv1_filters kernel_size dense2_units learning_rate	32 3 16 0.001	conv2_filters dense1_units dropout_rate	64 32 0.5
NN	hidden_layer_sizes learning_rate	(30, 15) adaptive	alpha	0.1
RF	max_depth min_samples_leaf	7 20	max_features n_estimators	sqrt 200
DT	max_depth min_samples_leaf	6 30	max_features min_samples_split	sqrt 2
KNN	pca__n_components knn__weights knn__leaf_size	0.75 uniform 50	knn__n_neighbors knn__metric	31 euclidean
SVM	C gamma	0.1 scale	kernel	rbf

5. Resultados

Esta seção apresenta os resultados obtidos na avaliação experimental dos modelos de aprendizado de máquina propostos como *proxies* de borda. Os experimentos iniciais avaliaram a capacidade de aproximação preditiva (a aptidão dos modelos em replicar matematicamente o consenso estabelecido na nuvem) em dois cenários: (i) utilizando a distribuição original dos dados desbalanceados; e (ii) utilizando balanceamento sintético via técnica SMOTE, visando analisar o impacto sobre as minorias.

No cenário sem balanceamento (Tabela 3), observa-se que os métodos de *Ensemble Learning* demonstraram alta eficácia no o papel de *proxy*. O classificador *Stacking* destacou-se com Acurácia Balanceada de 99,64% e MCC de 0,9952, indicando que o modelo foi capaz de mapear quase perfeitamente as regras de consenso estabelecidas pelo mecanismo da nuvem. O *Voting* obteve desempenho similar, confirmando a robustez da combinação de modelos.

A aplicação do SMOTE (Tabela 4) revelou ganhos expressivos para a CNN, que melhorou substancialmente sua sensibilidade à escassez de exemplos da classe minoritária. Em contraste, para os métodos intrinsecamente robustos (como *Stacking* e *Voting*), o balanceamento sintético não proporcionou ganhos significativos, sugerindo que a distribuição natural dos rótulos gerada pelo mecanismo de consenso é suficientemente clara para modelos de base arbórea.

Do ponto de vista de redes, o desempenho preditivo só tem valor se acompanhado de baixa latência (Figura 2). A DT despontou como o modelo mais veloz (3,26 ms por inferência), configurando-se como a solução limite para *hardware* de altíssima restrição. Contudo, o foco arquitetural recai sobre o *Stacking* e o *Voting*. Apesar da sobrecarga de

Tabela 3. Desempenho dos modelos

Modelo	Acc	Bal. Acc	F1-Macro	Recall	MCC	Log Loss	Overfit Gap
Stacking	**0,9970**	**0,9964**	**0,9958**	**0,9970**	**0,9952**	**0,0092**	0,0030
Voting	0,9960	0,9938	0,9947	0,9960	0,9936	0,0374	0,0034
RF	0,9660	0,9728	0,9553	0,9660	0,9477	0,0798	0,0046
CNN	0,9565	0,9639	0,9435	0,9565	0,9336	0,0842	0,0063
DT	0,9410	0,9519	0,9252	0,9410	0,9117	0,1489	0,0065
KNN	0,9435	0,9236	0,9236	0,9435	0,9096	0,2421	0,0083
NN	0,9400	0,9144	0,9182	0,9400	0,9046	0,2261	**0,0017**
SVM	0,9145	0,9039	0,8902	0,9145	0,8663	0,1979	0,0029

Tabela 4. Desempenho dos modelos com aplicação de SMOTE

Modelo	Acc	Bal. Acc	F1-Macro	Recall	MCC	Log Loss	Overfit Gap
Stacking	**0,9980**	**0,9978**	**0,9975**	**0,9980**	**0,9968**	**0,0217**	**0,0020**
Voting	0,9950	0,9936	0,9930	0,9950	0,9920	0,0359	0,0048
CNN	0,9920	0,9876	0,9899	0,9920	0,9872	0,0353	0,0047
RF	0,9730	0,9784	0,9641	0,9730	0,9582	0,0706	0,0062
DT	0,9605	0,9647	0,9480	0,9605	0,9390	0,1162	0,0100
NN	0,9470	0,9515	0,9325	0,9470	0,9183	0,1258	0,0018
KNN	0,9345	0,9324	0,9153	0,9345	0,8985	0,2762	0,0084
SVM	0,9135	0,9019	0,8887	0,9135	0,8646	0,2085	0,0177

operar múltiplos classificadores em conjunto, ambos registraram latências abaixo de 10 ms (respectivamente, $\approx 9,32$ ms e $\approx 7,86$ ms).

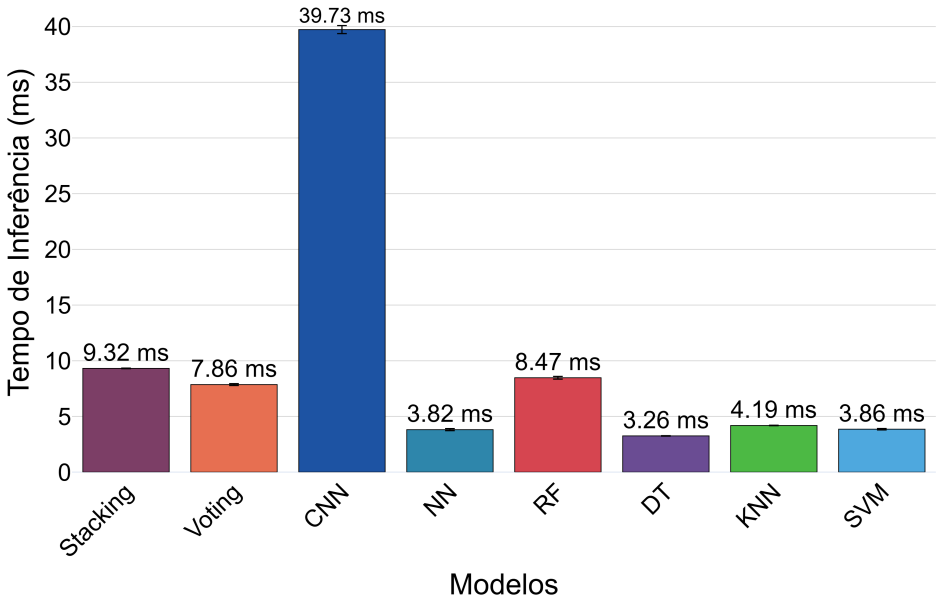

Figura 2. Comparativo de tempo de inferência.

Esse resultado é a validação primária da proposta, enquanto o módulo analítico na nuvem levaria dezenas de segundos para orquestrar as consultas CTI via internet, o

modelo de *ensemble* embarcado no *gateway* executa a inferência equivalente (com 99% de paridade em relação ao consenso) em menos de 10 milissegundos. Essa compressão de tempo é fundamental para intervenções proativas, como a inserção de conexões em *tarpit* antes mesmo do encerramento do *handshake* TCP.

Por fim, os testes temporais evidenciaram que modelos estáticos são insuficientes em cenários operacionais dinâmicos. Durante o período pré-*drift* (T_2), o modelo *Stacking* manteve alta eficácia, registrando acurácia de 98,5%. Contudo, ao ser submetido ao cenário de ataque *Zero-Day* simulado (T_{3A}), onde os principais pilares de inteligência do modelo (AbuseIPDB e VirusTotal) foram ofuscados pela tática de evasão, a acurácia decaiu para 79,9%.

Essa queda abrupta ultrapassou o limiar mínimo de desempenho estabelecido ($\tau = 0{,}92$), acionando imediatamente o mecanismo de alerta e retreinamento do AIOps. Durante a atualização do modelo na nuvem (incorporando os novos padrões anômalos de T_{3A}), o *Stacking* demonstrou a principal vantagem de sua natureza *ensemble*, o algoritmo ajustou automaticamente seus pesos internos, transferindo a relevância analítica (ou maior peso preditivo) para as fontes CTI de contingência, que não haviam sido comprometidas. Após a implantação do modelo atualizado na borda, verificou-se a recuperação da eficácia do sistema, atingindo 97,5% de acurácia no período subsequente (T_{3B}). Esses resultados comprovam que o ciclo fechado de AIOps (detecção de anomalia $\rightarrow$ retreinamento com ajuste dinâmico de características $\rightarrow$ atualização de borda) é a peça fundamental para garantir a resiliência de redes frente a fenômenos agressivos de *concept drift*.

6. Conclusão

Este trabalho apresentou e avaliou computacionalmente uma arquitetura de segurança baseada em AIOps para a classificação e mitigação de conexões IP, endereçando o gargalo de latência inerente à consulta de múltiplas fontes de Inteligência de Ameaças Cibernéticas através da redução do tempo de inferência isolado. A principal contribuição da proposta reside na utilização de modelos de aprendizado de máquina supervisionado não como mecanismos isolados de detecção, mas como *proxies* de baixa latência embarcados na borda da rede. Esses modelos demonstraram ser capazes de sintetizar e replicar o processamento analítico pesado (consenso multicritério) realizado na nuvem.

Os resultados experimentais confirmam a viabilidade operacional da abordagem. Modelos de *ensemble learning*, notadamente o *Stacking* e o *Voting*, alcançaram paridade superior a 99% com a *ground truth* da nuvem, mantendo tempos de inferência abaixo de 10 ms. Esse desempenho viabiliza intervenções proativas diretamente em *gateways* de rede, sem degradar a Qualidade de Serviço. Mais importante ainda, a simulação de *concept drift* evidenciou que modelos estáticos são vulneráveis a táticas de evasão (com a acurácia caindo para 79,9%), mas validou o ciclo fechado do AIOps, a arquitetura foi capaz de detectar a anomalia preditiva, reajustar dinamicamente o peso das fontes de CTI de contingência, e recuperar de forma autônoma a eficácia do sistema para 97,5% por meio do retreinamento.

Como trabalhos futuros, planejamos instanciar a arquitetura em *hardware* de rede programável (SmartNICs via eBPF/XDP) para endereçar as latências do plano de dados, validando o sistema sob tráfego orgânico contínuo. Para aprimorar a resiliência na borda, investigaremos políticas de *cache* para IPs desconhecidos, detecção de *concept drift* não

supervisionada e a integração de atributos comportamentais de fluxo. Por fim, exploraremos técnicas de *Explainable AI* (XAI) e estratégias *Human-in-the-Loop* (HITL) para refinar o ciclo AIOps e garantir governança diante de decisões e falsos positivos complexos.

Agradecimentos

Pesquisa parcialmente financiada pelo CNPq (Processos 305946/2025-0 e 405940/2022-0) e Coordenação de Aperfeiçoamento de Pessoal de Nível Superior – Brasil (CAPES) – Código de Financiamento 88887.954253/2024-00 e 88887.972043/2024-00.

Referências

Brito, M. L. L., Ferreira, M. C. M., Portela, A. L. C., and Gomes, R. L. (2025). Ai-based estimation of bandwidth availability for data offloading in edge-cloud computing. *IEEE Networking Letters*, pages 1–1.

Costa, M. A., Costa, Y. M., Almeida, Y. O., Cardoso, F. J., and Gomes, R. L. (2024). Connection management using automated firewall based on threat intelligence. In *Proceedings of the 2024 Latin America Networking Conference*, LANC '24, page 32–37, New York, NY, USA. Association for Computing Machinery.

Costa, W. L., Portela, A. L., and Gomes, R. L. (2021). Features-aware ddos detection in heterogeneous smart environments based on fog and cloud computing. *International Journal of Communication Networks and Information Security*, 13(3):491–498.

Dietterich, T. G. (2000). Ensemble methods in machine learning. In *Multiple classifier systems*, pages 1–15. Springer.

Ferreira, M. C., Ribeiro, S. E., Nobre, F. V., Linhares, M. L., Araújo, T. P., and Gomes, R. L. (2024). Mitigating measurement failures in throughput performance forecasting. In *2024 20th International Conference on Network and Service Management (CNSM)*, pages 1–7.

Gama, J. a., Žliobaitundefined, I., Bifet, A., Pechenizkiy, M., and Bouchachia, A. (2014). A survey on concept drift adaptation. *ACM Comput. Surv.*, 46(4).

Gomes, R. L., Bittencourt, L. F., and Madeira, E. R. M. (2013). A virtual network allocation algorithm for reliability negotiation. In *2013 22nd International Conference on Computer Communication and Networks (ICCCN)*, pages 1–7.

Gomes, R. L., Bittencourt, L. F., Madeira, E. R. M., Cerqueira, E. C., and Gerla, M. (2016). Software-defined management of edge as a service networks. *IEEE Transactions on Network and Service Management*, 13(2):226–239.

Huang, Y., Negrete, J., Wagener, J., et al. (2023). Graph neural networks and cross-protocol analysis for detecting malicious ip addresses. *Complex & Intelligent Systems*, 9:3857–3869.

Kuncheva, L. I. (2004). *Combining Pattern Classifiers: Methods and Algorithms*. John Wiley & Sons, Hoboken, NJ, USA.

Lazar, D., Cohen, K., Freund, A., Bartik, A., and Ron, A. (2021). Imdoc: Identification of malicious domain campaigns via dns and communicating files. *IEEE Access*, 9:45242–45258.

Nobre, F. V. J., Alves, D. O., Araujo, R. S., Campos, G. A., and Gomes, R. L. (2026). Risk classification of ip addresses using machine learning with weighted voting approach. In Rodrigues, L. A. and Oliveira, R., editors, *Dependable and Secure Computing*, pages 320–328, Cham. Springer Nature Switzerland.

Nobre, F. V. J., Silva, D. d. S., Ferreira, M. C. M. M., Brito, M. L. M. L., de Araújo, T. P., and Gomes, R. L. (2025). Time-weighted correlation approach to identify high delay links in internet service providers. *Journal of Internet Services and Applications*, 16(1):419–430.

Park, J., You, G., Ji, Y., and Youm, H. Y. (2024). Security requirements for fully automated ai systems to exercise and ensure the rights of data subjects. In *2024 19th Asia Joint Conference on Information Security (AsiaJCIS)*, pages 107–112.

Pimenta, I., Silva, D., Moura, E., Silveira, M., and Gomes, R. L. (2024). Impact of data anonymization in machine learning models. In *Proceedings of the 13th Latin-American Symposium on Dependable and Secure Computing*, pages 188–191.

Pimenta, I. A., Lee, M. H., Bittencourt, L. F., and Gomes, R. L. (2025). Adaptive privacy based on mutual information for machine learning in edge-cloud environments. *IEEE Networking Letters*, pages 1–1.

Portela, A. L. C., Ribeiro, S. E. S. B., Menezes, R. A., de Araujo, T., and Gomes, R. L. (2024). T-for: An adaptable forecasting model for throughput performance. *IEEE Transactions on Network and Service Management*, pages 1–1.

Potts, W. C. and Carver, C. (2024). Best practices implementing aiops in large organizations. In *2024 International Conference on Smart Applications, Communications and Networking (SmartNets)*, pages 1–5.

Siam, A. A., Alazab, M., Awajan, A., Hasan, M. R., Obeidat, A., and Faruqui, N. (2025). Ip safeguard–an ai-driven malicious ip detection framework. *IEEE Access*, 13:90249–90261.

Souza, M. S., Ribeiro, S. E. S. B., Lima, V. C., Cardoso, F. J., and Gomes, R. L. (2024). Combining regular expressions and machine learning for sql injection detection in urban computing. *Journal of Internet Services and Applications*, 15(1):103–111.

Spyros, A., Koritsas, I., Papoutsis, A., Panagiotou, P., Chatzakou, D., Kavallieros, D., Tsikrika, T., Vrochidis, S., and Kompatsiaris, I. (2025). Ai-based holistic framework for cyber threat intelligence management. *IEEE Access*, 13:20820–20846.

Usman, N., Usman, S., Khan, F., Jan, M. A., Sajid, A., Alazab, M., and Watters, P. (2021). Intelligent dynamic malware detection using machine learning in ip reputation for forensics data analytics. *Future Generation Computer Systems*, 118:124–141.

Wagner, T. D., Mahbub, K., Palomar, E., and Abdallah, A. E. (2019). Cyber threat intelligence sharing: Survey and research directions. *Computers Security*, 87:101589.

Yang, J. and Lim, H. (2021). Deep learning approach for detecting malicious activities over encrypted secure channels. *IEEE Access*, 9:39229–39244.

Yang, Y., Yang, S., Zhao, C., and Xu, Z. (2024). Telops: Ai-driven operations and maintenance for telecommunication networks. *IEEE Communications Magazine*, 62(4):104–110.