Arquitetura Híbrida de Aprendizado *Online* Adaptativa para Detecção de Ameaças *Zero-Day*

Marlon Brendo Ramos[1], **Rodrigo Sanches Miani**[1]

[1]Faculdade de Computação – Universidade Federal de Uberlândia (UFU)
Av. João Naves de Ávila, 2121 – 38.408-100 – Uberlândia – MG – Brazil

marlonbrendoramos@ufu.br, miani@ufu.br

Abstract. *In the context of increasingly complex and dynamic cyber threats, traditional detection models often prove insufficient against zero-day attacks due to their inability to adapt in real time and their susceptibility to concept drift. This article proposes a novel hybrid continual learning architecture designed not only to detect but also to autonomously assimilate new attack classes from data streams. Shifting away from reliance on offline retraining, the solution introduces an adaptive prequential mechanism that dynamically alternates between spatial and probabilistic metrics, ensuring the accurate identification of both volumetric and stealthy threats. Furthermore, the architecture incorporates a conservative auto-labeling process, enabling the model to evolve autonomously and learn new threats based on the consensus of its own predictions. Experimental results on the ERENO IEC61850 dataset validated the approach's efficacy, achieving an average F1-Score of 84,65% in zero-day class identification. The system's generalization capacity was further evaluated on the ML-EdgeIIoT dataset, where detection efficacy reached 84.15% , vastly outperforming baseline models and highlighting the viability of continual evolution in critical environments*

Resumo. *No contexto de ameaças cibernéticas cada vez mais complexas e dinâmicas, os modelos tradicionais de detecção falham contra ataques zero-day por não se adaptarem em tempo real e por serem vulneráveis ao concept drift, perdendo eficácia conforme os padrões mudam. Este artigo propõe uma nova arquitetura híbrida de aprendizado contínuo projetada não apenas para detectar, mas para assimilar autonomamente novas classes de ataques em fluxo de dados. Diferente de abordagens que dependem de retreinamentos offline, a solução introduz um mecanismo prequential adaptativo que alterna dinamicamente entre métricas espaciais e probabilísticas, garantindo a identificação precisa tanto de ameaças volumétricas quanto furtivas. Além disso, a arquitetura incorpora um processo de auto-rotulação conservadora, permitindo que o modelo evolua de forma autônoma e aprenda novas ameaças a partir do consenso de suas próprias predições. Resultados experimentais no dataset Ereno IEC61850 validaram a eficácia da abordagem, alcançando um F1-Score médio de 84,65% na identificação de classes zero-day. A capacidade de generalização do sistema foi avaliada no dataset ML-EdgeIIoT, onde a eficácia de detecção atingiu 84,15%, superando amplamente os modelos baseline e evidenciando a viabilidade da evolução contínua em ambientes críticos.*

1. Introdução

Em ambientes digitais altamente interconectados, a cibersegurança é fundamental para proteger operações contra ameaças cada vez mais sofisticadas e polimórficas. Sistemas de Detecção de Intrusão (IDS) baseados em aprendizado de máquina têm demonstrado grande potencial na mitigação desses riscos, especialmente ao superar a ineficiência das abordagens tradicionais baseadas puramente em assinaturas estáticas [Ahmed et al. 2025, Kikissagbe and Adda 2024, Ali et al. 2025]. Relatórios recentes evidenciam a gravidade do cenário: mais de 53% dos comprometimentos em larga escala em 2023 envolveram vulnerabilidades *zero-day* [Rapid7 Labs 2024], enquanto 75 falhas desse tipo foram exploradas ativamente em 2024 [Google Threat Intelligence Group 2025]. Esses dados reforçam a necessidade urgente de mecanismos de detecção adaptativos e proativos.

Apesar dos avanços significativos na literatura, as abordagens existentes ainda apresentam limitações. Modelos baseados em aprendizado em *batch* exigem retreinamentos periódicos e não conseguem se adaptar rapidamente a ataques inéditos [Pérez et al. 2017], tornando-se obsoletos diante da constante mudança do tráfego e do fenômeno de *Concept Drift* [Hindy et al. 2024, Khan et al. 2024]. Por outro lado, o aprendizado *online* ainda é pouco explorado em arquiteturas híbridas que integrem detecção de anomalias, classificação incremental multiclasse e mecanismos adaptativos [Nakıp and Gelenbe 2024].

Além da adaptação contínua, sistemas de detecção em redes reais precisam atender a restrições operacionais rigorosas. Em cenários com alto volume de tráfego, a análise dos pacotes deve ocorrer com baixa latência. Modelos com tempo de inferência elevado tornam-se inviáveis no tráfego em tempo real, pois podem gerar gargalos e permitir que ataques ocorram antes da emissão de alertas. Assim, manter um processamento rápido é fundamental para viabilizar a mitigação imediata de ameaças em fluxos contínuos de rede.

Para suprir essas lacunas, este trabalho propõe um modelo híbrido de aprendizado online voltado à detecção de ataques *zero-day*, buscando equilibrar desempenho, adaptabilidade e eficiência computacional. Dessa maneira, as principais contribuições deste artigo podem ser resumidas em três pilares: (i) Integração de Filtragem e Classificação em Fluxo: acopla a detecção inicial de anomalias via *Half-Space Trees* (HST) à classificação multiclasse incremental *AdaBoost* com *Hoeffding Trees* em tempo real; (ii) Identificação Dinâmica de Zero-Days: implementação de um mecanismo adaptativo de Reconhecimento de Conjunto Aberto (OSR) que alterna entre métricas espaciais como distância ao centróide e probabilísticas através da incerteza via entropia para identificar e isolar ameaças inéditas, e (iii) Evolução Autônoma via Auto-Rotulação: criação de um autorotulador conservador baseado em consenso preditivo que retroalimenta o modelo com instâncias de alta confiança, atualizando parâmetros e mitigando ativamente o *Concept Drift* em tempo real sem intervenção humana.

O restante do artigo está estruturado da seguinte maneira. A Seção 2 revisa os trabalhos relacionados e a Seção 3 apresenta a metodologia. A arquitetura do modelo, seu treinamento e avaliação contínua são detalhados nas Seções 4 e 5. Por fim, a Seção 6 discute os resultados experimentais e a Seção 7 conclui com as considerações finais e trabalhos futuros.

2. Trabalhos relacionados

A literatura recente em detecção de intrusões foca primariamente em maximizar a precisão preditiva utilizando aprendizado em lote (*batch learning*). Trabalhos como [Dai et al. 2024], [Nhlapo and Nkongolo 2024], [Sajid et al. 2024], [Chen et al. 2024] e [Ahmed et al. 2025] demonstram alta eficácia na classificação de ameaças *zero-day* usando *ensembles* e metamodelos robustos. Contudo, operam de forma estática e *offline*, inviabilizando a adaptação em tempo real ao *concept drift* sem a necessidade de retreinamentos custosos.

Em contrapartida, abordagens em fluxo contínuo ainda carecem de autonomia e completude. Touré et al. [Touré et al. 2024] utilizam *clustering* para sinalizar anomalias em tempo real, mas dependem de validação manual e não realizam classificação multiclasse. Por outro lado, [de Araújo Josephik et al. 2023] aplicam *Hoeffding Trees* em fluxo, mas restringem-se à detecção binária de ataques DDoS, sem abordar a descoberta autônoma de novas classes.

A Tabela 1 sintetiza o comparativo estruturado entre as soluções da literatura e a arquitetura proposta. Cada um dos campos representa dimensões centrais para a análise das abordagens. A coluna Detecção *Zero-Day* indica se o método é capaz de identificar ameaças inéditas, não presentes no conjunto de treinamento. Aprendizado em Fluxo informa se o modelo opera em cenário contínuo, processando instâncias sequencialmente sem depender de treinamento em lote. Abordagem OOD sinaliza a presença de mecanismos explícitos para detecção de amostras fora da distribuição conhecida ou de comportamentos anômalos. Classificação Multiclasse mostra se a solução distingue múltiplas categorias de ataques, em vez de apenas separar tráfego benigno e malicioso. Por fim, Atualização Autônoma indica se o sistema é capaz de incorporar novos conhecimentos sem intervenção manual, característica essencial para adaptação dinâmica ao *concept drift* e à detecção de novas ameaças.

Os resultados da revisão da literatura evidenciam a lacuna estrutural, a maioria das soluções ainda está presa à ideia de um mundo fechado ou depende de retreinamentos offline periódicos para lidar com o *Concept Drift*. Enquanto muitas abordagens recentes se concentram em detectores de novidade baseados em limiares fixos, que acabam se tornando rapidamente desatualizados ou gerando muitos falsos positivos em ambientes de tráfego dinâmico, o modelo proposto neste trabalho segue em outra direção.

Em contraste com a literatura, nossa arquitetura busca justamente enfrentar essa limitação. Ela separa a detecção espacial (HST) da classificação incremental e utiliza um seletor prequential que alterna dinamicamente entre métricas complementares como distância e entropia. Essa organização mostra que o sistema não foi pensado como um ajuste pontual para um cenário específico, mas como uma proposta mais robusta para lidar com a fragilidade estrutural dos modelos estáticos diante de ameaças zero-day contínuas.

3. Metodologia

Essa seção apresenta a descrição dos conjuntos de dados utilizados, modelos e testes realizados para a geração dos resultados.

Para validação do modelo híbrido proposto, foram utilizados dois *datasets* que representam cenários distintos. Para o primeiro cenário, o conjunto de dados Ereno

Tabela 1. Comparativo entre abordagens da literatura e o modelo proposto.

Referência	Detecção *Zero-Day*	Aprendizado em Fluxo	Abordagem OOD	Classificação Multiclasse	Atualização Autônoma
[Dai et al. 2024]	✓	✗	✗	✓	✗ *offline*
[Nhlapo and Nkongolo 2024]	✓	✗	✗	✓	✗ *offline*
[Sajid et al. 2024]	✓	✗	✗	✓	✗ *offline*
[Chen et al. 2024]	✓	✗	✗	✓	✗ *offline*
[Ahmed et al. 2025]	✓	✗	✗	✓	✗ *offline*
[Touré et al. 2024]	✓	✓	✓	✗ apenas binário	✗ manual
[de Araújo Josephik et al. 2023]	✗	✓	✗	✗ apenas DDoS	✗ *offline*
Modelo Proposto	✓	✓	✓ **centroide/ entropia**	✓	✓ **auto-rotulação**

IEC61850 IDS [Quincozes et al. 2024] foi selecionado. O conjunto representa dados de rede de subestações elétricas, contendo tráfego normal e diferentes tipos de ataques simulados. Os dados incluem protocolos industriais utilizados em subestações elétricas, como IEC 61850 e IEC 104. Por outro lado, para avaliar o modelo proposto em um outro domínio, utilizamos o *dataset* Edge-IIoT [Ferrag et al. 2022] que caracteriza um conjunto de dados voltado para detecções de intrusões em ambientes IoT industriais.

A escolha destes *datasets* rotulados alinha-se ao escopo da Computação Ubíqua e Pervasiva por refletir cenários críticos de *edge computing*, como *IoT* Industrial e subestações. Ao contrário de capturas genéricas, eles representam a real heterogeneidade dos sistemas pervasivos físicos e distribuídos, além dos rótulos necessários para garantir o rigor quantitativo e fidelidade na validação do modelo.

Os dados passaram por uma amostragem estratificada, equiparando-se o volume do *ML-EdgeIIoT* ao do *Ereno IEC61850* para comparações justas. Para contornar o desbalanceamento do tráfego das redes IoT, aplicou-se SMOTE no treinamento offline do *ML-EdgeIIoT*, assegurando representatividade às classes minoritárias. Atributos categóricos receberam *Ordinal Encoding*. Para o dataset *Ereno IEC61850*, foi selecionado o *MinMaxScaler*, enquanto para o *ML-EdgeIIoT*, utilizou-se o *StandardScaler*. Essa escolha foi definida com base em testes empíricos, que demonstraram ser a configuração mais eficiente para o desempenho de cada base de dados.

4. Arquitetura híbrida de aprendizado contínuo

O modelo híbrido adaptativo proposto consiste em uma arquitetura de quatro estágios, estruturada sob a ótica do Reconhecimento de Conjunto Aberto (OSR). O objetivo prático dessa base teórica é preparar o sistema para lidar com fluxos mais próximo das redes reais, superando a limitação dos classificadores tradicionais que falham diante de ataques inéditos.

Essa justificativa afasta o modelo de uma união puramente empírica de algoritmos, estabelecendo uma divisão clara e necessária de tarefas. Primeiro, o tráfego é filtrado na fase 1 via *HST* e os ataques conhecidos são mapeados na fase 2 via *AdaBoost*. Para descobrir as ameaças inéditas, na fase 3 a arquitetura exige a combinação de dois detectores complementares, a distância espacial identifica distorções de volume (ataques volumétricos), enquanto a entropia avalia a incerteza do modelo para capturar ameaças furtivas. É exatamente essa dupla checagem que forma o alicerce seguro para a evolução autônoma do sistema.

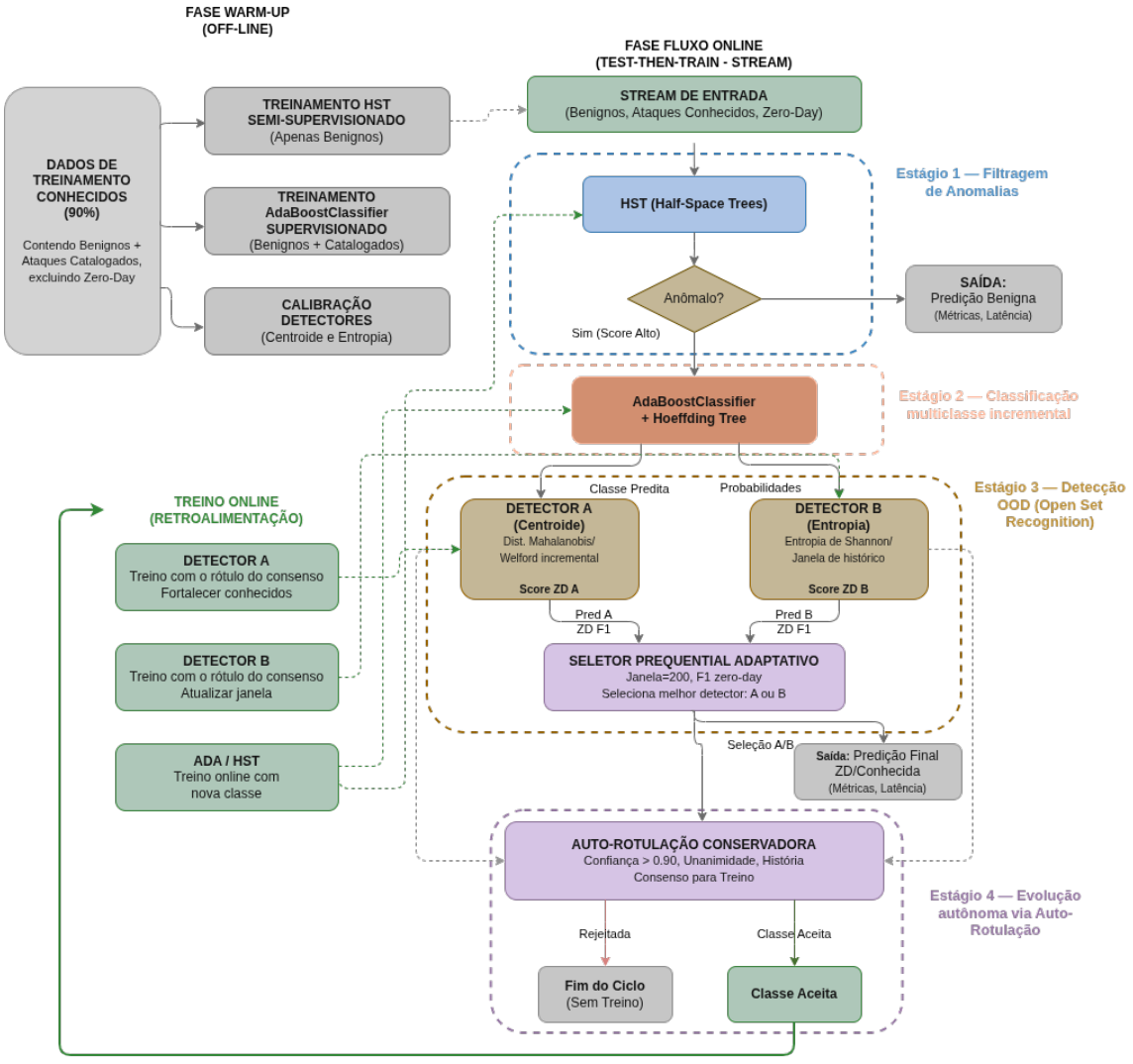

Figura 1. Arquitetura do modelo híbrido adaptativo proposto para detecção de ataques *Zero-Day*.

Conforme ilustrado na Figura 1, a arquitetura é estruturada em quatro etapas. Inicialmente, um detector de anomalias analisa fluxos de rede em tempo real para identificar desvios de comportamento. Em seguida, as instâncias consideradas anômalas são encaminhadas a um classificador multiclasse, responsável por inferir o rótulo com base nas classes previamente aprendidas. Na terceira etapa, um seletor *prequential* com janela deslizante avalia a predição e aciona o mecanismo de OSR com maior acurácia recente

para a classe em questão, alternando entre métodos baseados em distância e em entropia. No contexto de aprendizado em fluxo, a avaliação *prequential* (*predictive sequential*) consiste em testar o modelo em cada nova instância antes de utilizá-la para atualização, permitindo monitorar continuamente seu desempenho ao longo do fluxo de dados. O mecanismo da terceira etapa confirma a classificação inicial ou a rejeita, sinalizando uma potencial ameaça *zero-day*. Por fim, um auto-rotulador conservador avalia o consenso dessas predições para retroalimentar a arquitetura em tempo real, viabilizando o aprendizado contínuo da nova ameaça sem intervenção humana. As etapas a seguir apresentam, de forma detalhada, cada estágio do modelo.

4.1. Fase 1: Filtragem e *Warm-up*

A escolha específica do algoritmo *Half-Space Trees* (HST) em detrimento de alternativas tradicionais como *Isolation Forest* ou *One-Class SVM* fundamenta-se nas restrições operacionais de redes reais. Enquanto métodos baseados em kernel ou árvores estáticas exigem alto custo de memória ou retreinamentos em lote (*batch*), o HST possui complexidade computacional constante para tempo e espaço por instância processada. Essa característica garante o isolamento rápido de anomalias espaciais sem comprometer a latência da rede.

Para materializar nossa primeira contribuição, os dois primeiros estágios do modelo operam em conjunto para processar o fluxo de dados em tempo real sem retreinamentos offline. Para isso, antes de ser exposto ao *stream* completo, o modelo de detecção de anomalias passa por uma fase de *warm-up*, na qual aprende exclusivamente a partir de amostras da classe normal. Essa etapa inicial é fundamental para que o modelo construa uma representação do tráfego normal. Em seguida, todos os dados de treino passam pelo HST para gerar *scores*, e o *threshold* ótimo é encontrado avaliando 300 candidatos pelo critério de máximo F1 binário. Ademais, expor o HST a ataques logo durante a inicialização contaminaria a distribuição de referência, degradando a capacidade de detectar anomalias, para isso, o *warm-up* utiliza o conjunto de treino supervisionado para estabelecer uma representação estável da classe normal antes da inferência.

4.2. Fase 2: Classificação Inicial

Os fluxos anômalos da etapa anterior seguem para o segundo estágio, um classificador AdaBoost online com *Hoeffding Trees*. Essa configuração garante processamento contínuo e baixo uso de memória, elevando a precisão do mapeamento multiclasse em padrões complexos sem comprometer o desempenho em tempo real.

O objetivo nesta etapa é enquadrar a instância nas classes de ataque já conhecidas. Contudo, por operar restrito aos rótulos que aprendeu, o AdaBoost pode sugerir que a amostra é normal. Como o HST já confirmou a anomalia, surge uma contradição direta. O sistema não assume essa divergência como um erro definitivo, em vez disso, encaminha o rótulo sugerido e as probabilidades para a Fase 3, onde a real aderência será inspecionada para que, na Fase 4, a auto-rotulação valide a predição por consenso e alta confiança.

4.3. Fase 3: Detecção de Novidades (*OOD*)

Para identificar dinamicamente ameaças *zero-day*, a terceira etapa da arquitetura introduz uma validação *Out-of-Distribution* (OOD). Este módulo audita o rótulo inferido pelo

AdaBoost com *Hoeffding Trees* e verifica sua compatibilidade à distribuição dos ataques já conhecidos. Esse mecanismo mitiga o excesso de confiança do classificador e impede que o sistema rotule ataques inéditos como ameaças catalogadas. Para garantir essa robustez, a solução integra dois detectores em um *ensemble*, de modo que um algoritmo compense ativamente as limitações do outro.

4.3.1. Detector A (Centroide)

O objetivo desse detector é capturar os ataques volumétricos, ou seja, ataques de *flooding* e varredura como por exemplo: DDoS, DoS, TCP SYN *Flood*, UDP *Flood*, *Spoofing*, entre outros. A premissa é que tais ameaças causam distorções abruptas e em larga escala nos atributos estatísticos de fluxo, essas instâncias são projetadas para regiões periféricas bem distantes do centro de massa representado pelo tráfego legítimo. Dessa maneira o detector baseado na distância de *Mahalanobis* simplificada opera calculando o afastamento espacial da amostra em relação ao centroide da classe sugerida pelo classificador do estágio anterior.

4.3.2. Detector B (Entropia)

Por sua vez, o detector baseado em entropia atua com foco em ataques mais furtivos como *Slowloris*, *Low-rate* DDos e FDIA, que tendem a apresentar padrões menos evidentes no tráfego. Diferente dos ataques volumétricos, essas ameaças são projetadas para mimetizar o tráfego legítimo ou mesclar múltiplas assinaturas. Consequentemente, tais instâncias tendem a ser projetadas próximas ás fronteiras de decisões do classificador do estágio anterior. Para mensurar a incerteza, o detector não avalia os atributos da rede, mas sim a distribuição de probabilidades gerada pela camada de saída do *AdaBoost*. Matematicamente, a incerteza da predição é calculada através da entropia de *Shannon*. Em outras palavras, se as porcentagens de probabilidades vindas do *AdaBoost* estiverem muito equilibradas e parecidas, significa confusão. Confusão significa alta entropia, podendo ser um *zero-day* mais furtivo.

4.3.3. Seletor *Prequential* Adaptativo

A arquitetura descarta métodos estáticos de votação, como votação majoritária ou média e emprega um seletor *prequential* adaptativo para consolidar a saída do *ensemble* OOD. Este componente mantém uma janela deslizante com as últimas 200 instâncias processadas e avalia dinamicamente o desempenho do *F1-Score* recente de ambos os detectores. Em seguida, o sistema delega a decisão final estritamente ao algoritmo que apresentar a maior competência local naquele histórico imediato. Essa alternância contínua entre métricas espaciais e probabilísticas adapta o modelo em tempo real e mitiga diretamente os impactos do *Concept Drift*.

4.4. Evolução Autônoma via Auto-Rotulação Conservadora

Finalmente, para viabilizar a evolução autônoma do modelo e atenuar o *concept drift*, a arquitetura adota o paradigma *Test-Then-Train*. O mecanismo de auto-rotulação valida

e reinjeta uma nova amostra no sistema apenas se houver consenso preditivo absoluto. Na prática, isso exige que os dois detectores OOD concordem na classificação, que o *AdaBoost* apresente um nível de confiança superior a 90%, e que a amostra pertença a uma janela deslizante recente. Uma vez validada, a instância retroalimenta imediatamente a arquitetura, o *AdaBoost* e o HST ajustam os limites de suas fronteiras de decisão, o detector OOD de centroide desloca seu centro matemático, via algoritmo de Welford, e o detector OOD de entropia calibra seu limiar de tolerância à incerteza. Esse *feedback* contínuo e rigoroso garante que o modelo aprenda as sutis alterações de comportamento da rede em tempo real.

Além disso, o mecanismo também mitiga o erro acumulativo, visto que, a autorotulação conservadora proposta impede que predições ambíguas ou incorretas retroalimentem a árvore de decisão. Por outro lado, ainda existe a possibilidade teórica de degradação a longo prazo por ataques adversariais desenhados para burlar os limiares de confiança, a natureza incremental das Hoeffding Trees faz com que o impacto de instâncias ruidosas isoladas tenda a ser diluído ao longo do tempo, conforme um volume maior de exemplos legítimos vai sendo incorporado ao modelo.

5. Metodologia de Treinamento Inicial e Avaliação Contínua

Para garantir a reprodutibilidade, os experimentos foram conduzidos em ambiente Python 3.13, utilizando as bibliotecas *NumPy* (2.4.1), *Pandas* (2.3.3), *River* (0.23.0) e *Scikit-Learn* (1.8.0), em um sistema equipado com processador Intel Core i7-13620H de 13ª geração e 16GB de RAM. O código-fonte está disponível em: `https://github.com/MarlonBrendonx/Hybrid-Online-Model-Zero-Day`.

Para avaliar a detecção de intrusões inéditas, adotou-se um rodízio sistemático, isolando iterativamente cada classe de ataque como *zero-day* e ocultando-a totalmente do treinamento. Embora essa abordagem seja o padrão na literatura para permitir comparações justas, ela apresenta limitações práticas, pois amostras extraídas de um mesmo conjunto de dados tendem a compartilhar padrões e características da rede de origem. A arquitetura proposta, contudo, atenua esse viés ao empregar verificações complementares na Fase 3. O sistema não depende apenas da distância dos dados em relação ao tráfego normal, caso um ataque zero-day seja muito similar ao tráfego conhecido e passe pelo cálculo de distância, ele inevitavelmente gerará um alto nível de dúvida na tomada de decisão do classificador, sendo capturado e isolado pelo detector de incerteza.

A etapa de *warm-up* utilizou os primeiros 90% do tráfego conhecido, o filtro HST mapeou o comportamento normal, enquanto o *AdaBoost* e os detectores OOD calibraram suas fronteiras com as assinaturas catalogadas.

Na fase de testes, simulou-se uma única execução em fluxo contínuo combinando os 10% restantes do tráfego conhecido com a classe *zero-day*. Para evitar o desbalanceamento, as amostras do ataque inédito sofreram *undersampling*, tendo seu volume limitado ao tamanho da classe majoritária do conjunto de teste. Para garantir a validade do experimento, o fluxo não sofreu embaralhamento aleatório. Essa preservação da ordem temporal é essencial para avaliar a adaptação ao *concept drift*, visto que ataques reais ocorrem cronologicamente e em rajadas. A avaliação seguiu o paradigma *Test-then-Train*, ou seja, instâncias rotuladas como anomalias de alta confiança atualizam os modelos imediatamente. Assim, as métricas de eficácia global reportadas representam a média de

desempenho da arquitetura após a conclusão de todos os ciclos de rodízio.

Como baseline, foi utilizado um AdaBoost online com Hoeffding Trees, treinado sem algumas classes para simular o cenário zero-day. A detecção de ameaças zero-day baseia-se apenas em uma heurística de confiança, quando a maior probabilidade prevista para as classes conhecidas é inferior a 70%, a amostra é considerada inédita. Esse limiar foi definido empiricamente e serve como referência direta para avaliar os ganhos da abordagem proposta.

6. Resultados

A Tabela 2 demonstra o desempenho detalhado da arquitetura no dataset *Ereno IEC61850*. O alcance de um F1-Score médio global de 84,65% para as classes *zero-day* não é apenas um incremento quantitativo sobre o baseline estático, mas a validação empírica da nossa primeira e segunda contribuições. Ao integrar a arquitetura híbrida de aprendizado contínuo com o mecanismo de identificação dinâmica, o seletor *prequential* conseguiu alternar com sucesso entre a detecção de ataques volumétricos e furtivos em tempo real. Essa sinergia fica evidente no estudo de ablação 3, onde a remoção de qualquer um dos detectores OOD causa uma queda drástica na eficácia de detecção de *zero-days*. Isso indica que o bom desempenho do modelo deriva da sua capacidade de adaptação dinâmica, e não do mérito de um único algoritmo isolado.

Tabela 2. Desempenho de Detecção *Zero-Day* para o Ereno IEC61850

Classe *Zero-Day* (Ataque)	*Baseline* F1 ZD	Proposto F1 ZD	Ganho (Δ ZD)	Latência *Base.* (μs)	Latência Prop. (μs)	Auto-Rotulação
Grayhole	0.4785	**0.9389**	+0.4604	5640.5	8223.7	5.8%
High_StNum	0.5213	**0.7495**	+0.2282	3317.4	9180.7	5.1%
Injection	0.4377	**0.9189**	+0.4813	3199.2	8941.2	6.7%
Inverse Replay	0.7195	**0.8739**	+0.1544	2949.3	8633.6	4.7%
Masquerade Fake Fault	0.6178	**0.8359**	+0.2181	3598.6	12797.1	6.7%
Masquerade Fake Normal	0.4077	**0.7653**	+0.3576	3830.3	7591.2	13.1%
Poisoned High Rate	0.4149	**0.7828**	+0.3678	3984.8	8243.5	12.9%
Random Replay	0.5293	**0.9066**	+0.3773	3470.5	10139.5	4.2%
Média Global	**0.5159**	**0.8465**	**+0.3306**	**3748.8**	**9218.8**	**7.4%**

Notas: Classe Zero-Day identifica o ataque omitido durante o treinamento para validação de ameaças desconhecidas; ***Baseline*** representa o modelo de referência com regras estáticas; **Proposto** e **Ganho** detalham a performance da arquitetura híbrida e seu incremento percentual; **Latência** reflete o tempo computacional de inferência por amostra; **Auto-Rotulação** indica a taxa de amostras incorporadas ao conhecimento do modelo via auto-rotulação.

Para assegurar que o ganho de desempenho deriva de cada estágio da arquitetura proposta, e não da limitação do modelo base, o estudo de ablação descrito na Tabela 3 foi desenhado para atuar como um conjunto rigoroso de *baselines* internos comparativos. A arquitetura foi testada com alguns módulos desativados para simular versões parciais do modelo: (i) a remoção do mecanismo de evolução autônoma e auto-rotulação contínua; (ii) a desativação do detector baseado em centroides para a detecção de classe aberta; (iii) a remoção do detector de incerteza baseado em entropia; e (iv) a ausência do módulo de filtragem de anomalias.

Os testes evidenciaram que a queda drástica na detecção de ameaças *zero-day* ocorre sempre que o sistema é reduzido a um modelo tradicional estático ou de métrica única. Isso mostra empiricamente que o desempenho superior do modelo não é mérito de um algoritmo isolado, mas sim do mecanismo *prequential* que alterna dinamicamente as métricas e da capacidade de retroalimentação em tempo real.

Tabela 3. Resumo Comparativo Entre o Modelo Proposto e Estudo de Ablação

Arquitetura / Cenário	F1-Score Médio	Ganho (△) vs. *Baseline*	Latência Média (μs)	Taxa de Auto-Rotulação
Híbrido (Proposto)	**0.8465**	**+0.3306**	**9218.8**	**7.4%**
Híbrido *sem Auto-Rotulação*	0.6117	+0.0977	8744.9	—
Híbrido *sem CentroideOSR*	0.5121	-0.0019	8215.8	6.4%
Híbrido *sem EntropiaOSR*	0.2307	-0.2833	4030.0	6.0%
Híbrido *sem HST*	0.7222	+0.2082	8304.3	6.0%

Notas: O **Híbrido** representa a arquitetura final proposta. As linhas subsequentes demonstram a degradação do desempenho (F1-Score) e a variação da latência ao remover módulos individuais, evidenciando a dependência do sistema em relação ao detector de entropia, o filtro espacial de centroides, a auto-rotulação e o módulo HST.

Para verificar se o modelo proposto não se ajustou excessivamente às particularidades de uma rede, testamos a arquitetura híbrida em outro *dataset*, que não introduziu apenas novos protocolos, mas também vetores de ataque inéditos para o modelo. Como evidenciado na Tabela 4, os resultados comprovam, na prática, que o alto desempenho da solução não está associado a memorização de assinaturas, ou para um domínio específico. Pelo contrário, demonstra que o mecanismo de aprendizado contínuo proposto conseguiu capturar a essência do problema em questão.

Tabela 4. Desempenho de Detecção *Zero-Day* para o *dataset* Edge/IoT

Classe *Zero-Day* (Ataque)	*Base.* F1 ZD	Prop. F1 ZD	Ganho (△)	Lat. *Base.* (μs)	Lat. Prop. (μs)	Auto-Rot.
Port Scanning	0.2219	**0.9421**	+0.7202	2285.4	25749.2	28.0%
Ransomware	0.3772	**0.8352**	+0.4580	4508.4	25577.8	25.4%
DDoS TCP	0.4822	**0.8866**	+0.4044	1926.0	27201.3	25.4%
DDoS ICMP	0.2559	**0.8616**	+0.6057	2239.6	17838.0	21.7%
DDoS UDP	0.5571	**0.8352**	+0.2781	1857.4	31052.3	21.7%
Uploading	0.4753	**0.8866**	+0.4114	5709.7	26403.5	22.1%
Password	0.6072	**0.8394**	+0.2321	4874.7	22060.1	25.1%
SQL Injection	0.5700	**0.8161**	+0.2461	4457.1	19816.4	26.0%
MITM	0.4679	**0.8331**	+0.3652	6233.6	18905.3	22.3%
Fingerprinting	0.4296	**0.8463**	+0.4168	1457.1	25403.6	12.2%
DDoS HTTP	0.5680	**0.7912**	+0.2232	4108.6	27984.8	27.9%
XSS	0.5108	**0.8418**	+0.3310	4346.4	26215.7	27.3%
Backdoor	0.2795	**0.6567**	+0.3772	2732.5	27921.1	19.5%
Vulnerability Scanner	0.4763	**0.9097**	+0.4334	2206.6	25624.6	26.7%
Média Global	**0.4507**	**0.8415**	**+0.3908**	**3485.2**	**24839.6**	**23.7%**

O estudo de ablação evidencia que o alto desempenho da arquitetura resulta da função específica e indispensável de cada um de seus módulos. O detector baseado em entropia atua como o alicerce da arquitetura, sua remoção faz o F1-Score cair drasticamente, provando ser o mecanismo central para mensurar a incerteza e separar o tráfego conhecido das novidades, o que justifica seu maior impacto na latência. Complementar a ele, o detector baseado em centroide é o diferencial crítico que permite à arquitetura superar as abordagens tradicionais, sem esse mapeamento espacial, a vantagem sobre o *baseline* desaparece por completo. Adicionalmente, o HST atua filtrando anomalias e ruídos de forma eficiente, cuja ausência degrada a precisão geral. Por fim, ao desativar a auto-rotulação, o desempenho cai significativamente, confirmando que aprender continuamente com novos padrões é essencial para manter a eficácia e a robustez do sistema ao longo do tempo

No entanto, o estudo apresenta limitações inerentes ao uso de *datasets* controlados (*testbeds*). Nesses ambientes estáticos, a dinâmica de tráfego é mais previsível do que em redes reais, o que restringe a análise de generalização do modelo frente a ruídos não catalogados em produção. Além disso, existe o risco de contaminação do tráfego normal, se um ataque inédito conseguir mimetizar perfeitamente o comportamento legítimo, o sistema pode internalizá-lo por engano. Atualmente, a arquitetura também agrupa as novas ameaças em uma categoria generalista de *zero-day*, sem separá-las em famílias específicas.

7. Conclusões

Este trabalho supera as limitações de modelos estáticos ao consolidar três pilares: Arquitetura Híbrida de Aprendizado Contínuo, Identificação Dinâmica de *zero-days* e Evolução Autônoma. Os resultados atestam que o modelo não apenas detecta, mas também assimila novos ataques em tempo real, mitigando a dependência de retreinamentos offline para evoluir organicamente com o fluxo de rede.

Embora o *pipeline* híbrido proposto demonstre avanços significativos na superação das limitações dos modelos convencionais em modo *batch*, a detecção de ameaças *zero-day* em ambientes de aprendizado *online* ainda apresenta obstáculos complexos. A partir dos resultados, delineiam-se as seguintes direções para trabalhos futuros: (i) Resiliência a mudanças de conceito: explorar mecanismos de decaimento dinâmico ou janelas de memória adaptativas para o auto-rotulador, evitando o esquecimento catastrófico; (ii) Clusterização automática de novas ameaças: implementar algoritmos de agrupamento *online* integrados aos detectores de novidade para categorizar anomalias em famílias de ataques específicas, superando a atual rotulação generalista; (iii) Estratégia de atualização restritiva: investigar mecanismos onde o auto-rotulador retroalimente o *HST* exclusivamente com instâncias classificadas com alta confiança como tráfego normal, impedindo que o filtro de anomalias adapte gradualmente sua estrutura caso uma ameaça inédita se torne persistente e (iv) avaliar a proposta em diferentes *datasets*.

Agradecimentos

O apoio a esta pesquisa foi fornecido pelos projetos INCT-IACiber financiado pelo CNPq processo nº 408432/2024-1 e CYBERGUARD: Inteligência Artificial para Detecção e Resposta de Ameaças Cibernéticas financiado pelo CNPq process nº 409743/2025-9.

Declaração sobre o uso de Inteligência Artificial

Os autores declaram o uso de modelos de linguagem de grande porte (LLMs), como ChatGPT e Claude, exclusivamente como ferramentas de apoio à fluidez e aprimoramento da redação de parágrafos deste artigo. Todas as ideias, interpretações, análises e conclusões apresentadas são de responsabilidade integral dos autores.

Referências

Ahmed, U., Jiangbin, Z., Khan, S., and Sadiq, M. T. (2025). Consensus hybrid ensemble machine learning for intrusion detection with explainable ai. *Journal of Network and Computer Applications*, 235:104091.

Ali, M. L., Thakur, K., Schmeelk, S., Debello, J., and Dragos, D. (2025). Deep learning vs. machine learning for intrusion detection in computer networks: A comparative study. *Applied Sciences*, 15(4):1903.

Chen, Z., Simsek, M., Kantarci, B., Bagheri, M., and Djukic, P. (2024). Machine learning-enabled hybrid intrusion detection system with host data transformation and an advanced two-stage classifier. *Computer Networks*, 250:110576.

Dai, Z., Por, L. Y., Chen, Y.-L., Yang, J., Ku, C. S., Alizadehsani, R., and Pławiak, P. (2024). An intrusion detection model to detect zero-day attacks in unseen data using machine learning. *PloS one*, 19(9):e0308469.

de Araújo Josephik, J. G. A., Siqueira, Y., Machado, K. G., Terada, R., dos Santos, A. L., Nogueira, M., and Batista, D. M. (2023). Applying hoeffding tree algorithms for effective stream learning in iot ddos detection. In *2023 IEEE Latin-American Conference on Communications (LATINCOM)*, pages 1–6. IEEE.

Ferrag, M. A., Friha, O., Hamouda, D., Maglaras, L., and Janicke, H. (2022). Edge-iiotset: A new comprehensive realistic cyber security dataset of iot and iiot applications: Centralized and federated learning.

Google Threat Intelligence Group (2025). Hello 0-days, my old friend: A 2024 zero-day exploitation analysis. Technical report, Google. Document released April 2025.

Hindy, H. et al. (2024). Unveiling machine learning strategies and considerations in intrusion detection systems: a comprehensive survey. *Frontiers in Computer Science*, 6.

Khan, A. et al. (2024). Impact of machine learning on intrusion detection systems for the protection of critical infrastructure. *Information*, 16(7):515.

Kikissagbe, B. R. and Adda, M. (2024). Machine learning-based intrusion detection methods in iot systems: A comprehensive review. *Electronics*, 13(18):3601.

Nakıp, M. and Gelenbe, E. (2024). Online self-supervised deep learning for intrusion detection systems. *IEEE Transactions on Information Forensics and Security*.

Nhlapo, S. J. and Nkongolo, M. N. W. (2024). Zero-day attack and ransomware detection. *arXiv preprint arXiv:2408.05244*.

Pérez, J. L. R., Ribeiro, B., and Ortiz, K. H. (2017). A comparison of algorithms for intrusion detection on batch and data stream environments. *arXiv preprint arXiv:1701.00893*.

Quincozes, S. E., Albuquerque, C., Passos, D., and Mosse, D. (2024). ERENO: A Framework for Generating Realistic IEC–61850 Intrusion Detection Datasets for Smart Grids . *IEEE Transactions on Dependable and Secure Computing*, 21(04):3851–3865.

Rapid7 Labs (2024). The 2024 attack intelligence report. Technical report, Rapid7, Inc. GlobeNewswire, 21 May 2024.

Sajid, M., Malik, K. R., Almogren, A., Malik, T. S., Khan, A. H., Tanveer, J., and Rehman, A. U. (2024). Enhancing intrusion detection: a hybrid machine and deep learning approach. *Journal of Cloud Computing*, 13(1):123.

Touré, A., Imine, Y., Semnont, A., Delot, T., and Gallais, A. (2024). A framework for detecting zero-day exploits in network flows. *Computer Networks*, 248:110476.