Uma Ferramenta para Classificação de Tráfego Cifrado em VPNs sem Inspeção da Carga Útil dos Pacotes

Enzo Basaldella[1], Dalbert Matos Mascarenhas[1], Igor Monteiro Moraes[2]

[1] Centro Federal de Educação Tecnológica Celso Suckow da Fonseca (CEFET/RJ)
Petrópolis – RJ – Brasil

[2]Laboratório MidiaCom – IC/TCC/PGC
Universidade Federal Fluminense (UFF)
Niterói – RJ – Brasil

enzobasaldella@gmail.com, dalbert.mascarenhas@cefet-rj.br, igor@ic.uff.br

Resumo. *A adoção crescente de VPNs e criptografia de ponta a ponta torna ineficazes abordagens baseadas em inspeção de carga útil dos pacotes. Este trabalho propõe e valida uma ferramenta para classificação de tráfego cifrado em VPNs, operando sobre atributos extraídos do fluxo de pacotes cifrados, sem necessidade de decifração. A solução implementa um pipeline incremental para coleta automatizada, auditoria formal, extração de 80 atributos e treinamento de modelos supervisionados. A base de dados gerada reúne mais de 25.000 janelas temporais, extraídas de 800 sessões de oito serviços em duas macrocategorias. Os classificadores atingem F1-Macro de até 98,8% na classificação binária e 97,2% na multiclasse, evidenciando a eficácia da ferramenta proposta.*

Abstract. *The growing adoption of VPNs and end-to-end encryption makes packet-payload inspection approaches ineffective. This paper proposes and validates a tool for VPN encrypted traffic classification, operating on attributes extracted from encrypted packet flows without decryption. The solution implements an incremental pipeline for automated collection, formal auditing, extraction of 80 attributes, and training of supervised models. The generated dataset contains more than 25,000 time windows extracted from 800 sessions of eight services in two macro-categories. The classifiers achieve Macro-F1 scores of up to 98.8% in binary classification and 97.2% in multiclass classification, showing the effectiveness of the proposed tool.*

1. Introdução

A criptografia consolidou-se como elemento estrutural da comunicação em rede contemporânea. A adoção massiva de protocolos como TLS 1.3 e QUIC, associada ao uso crescente de VPNs, reduziu a visibilidade de monitores intermediários sobre o conteúdo das comunicações e limitou a efetividade de técnicas baseadas em inspeção profunda da carga útil dos pacotes [Shen et al. 2023, Razooqi and Pekár 2025]. No entanto, a inacessibilidade da carga útil não elimina a necessidade operacional de inferir padrões de uso em redes: operadoras precisam distinguir classes de tráfego para políticas de qualidade de serviço e planejamento de capacidade [Nguyen and Armitage 2008]; equipes

de segurança precisam identificar comportamentos anômalos, comunicações maliciosas e exfiltração de dados a partir de atributos estatísticos do fluxo de pacotes, sem depender de acesso ao conteúdo cifrado nem violar a privacidade [Anderson and McGrew 2017]; e a ausência de procedimentos de classificação metodologicamente controlados dificulta a comparação entre abordagens e limita a reprodutibilidade dos resultados reportados [Razooqi and Pekár 2025].

O uso de VPNs amplia esse desafio. Ao encapsular fluxos de diferentes aplicações em um único túnel cifrado, o tráfego de serviços distintos torna-se indistinguível por análise de porta ou endereçamento: toda a diversidade de serviços é vista pelo observador como um fluxo homogêneo [Kotak et al. 2025, Razooqi and Pekár 2025]. Nesse cenário, a classificação depende de metadados de tráfego, isto é, de atributos estatísticos extraídos do próprio fluxo de pacotes cifrados, tornando o aprendizado de máquina uma abordagem eficaz e recorrente na literatura.

Embora trabalhos recentes tenham demonstrado a viabilidade da classificação de tráfego cifrado com aprendizado de máquina [Lotfollahi et al. 2020], parte da literatura ainda concentra maior atenção na etapa de modelagem do que no controle de qualidade e na rastreabilidade da cadeia experimental, da coleta à avaliação. Essa lacuna dificulta a reprodutibilidade, a comparação entre estudos e a incorporação sistemática de novos pacotes de dados. O problema é especialmente relevante em conjuntos de dados construídos por janelamento temporal, nos quais a separação inadequada entre treino e teste pode introduzir vazamento de dados e inflar artificialmente as métricas reportadas [Kaufman et al. 2012, Zhao et al. 2025]. Além disso, bases de referência amplamente reutilizadas apresentam inconsistências registradas que dificultam comparações diretas [Draper-Gil et al. 2016].

Nesse contexto, este artigo propõe e valida uma ferramenta para classificação de tráfego cifrado em VPNs, estruturada como um pipeline incremental e reproduzível, com controle metodológico registrado em cada etapa da cadeia experimental, da coleta à avaliação dos resultados. As contribuições principais são:

- cenário experimental controlado com separação explícita entre nó gerador de tráfego e o nó monitor, que observa apenas o túnel cifrado, aproximando a coleta da perspectiva de um monitor de rede intermediário;
- acervo de 800 sessões de captura auditadas cobrindo oito serviços em duas macrocategorias (*web browsing* e *streaming*), com variabilidade comportamental controlada e critérios formais de qualidade aplicados antes do janelamento;
- processo de construção da base de dados, com extração de 80 atributos estatísticos por janela temporal, com possibilidade de expansão para novas capturas;
- protocolo de avaliação com separação estrita por sessão, reduzindo o risco de vazamento entre treino e teste e garantindo generalização e aprendizado real.

O restante do artigo está organizado da seguinte forma. A Seção 2 revisa os trabalhos relacionados. As Seções 3 e 4 descrevem o cenário experimental e a metodologia, respectivamente. A Seção 5 apresenta os resultados, e a Seção 6 conclui o artigo.

2. Trabalhos Relacionados

A análise de tráfego cifrado com aprendizado de máquina é uma linha consolidada e bem documentada. Levantamentos recentes organizam o campo segundo objetivos

de análise, formas de representação do tráfego e famílias de modelos, confirmando que fluxos de pacotes cifrados preservam padrões estatísticos e temporais suficientes para inferência mesmo sem acesso à carga útil [Shen et al. 2023]. No recorte específico de túneis VPN, Razooqi e Pekár [Razooqi and Pekár 2025] identificam três frentes centrais na literatura: detecção da presença de VPN, identificação do protocolo VPN e identificação da aplicação encapsulada. Esta última frente é diretamente relevante para o presente trabalho.

Em 2016, Draper-Gil et al. [Draper-Gil et al. 2016] introduziram o ISCXVPN2016, base de dados de referência amplamente adotada para classificação de tráfego VPN. Diferentemente deste trabalho, o ISCXVPN2016 é organizado em fluxos rotulados por categoria e tipo de aplicação, descritos por atributos temporais de fluxo.

A abordagem Deep Packet [Lotfollahi et al. 2020] demonstrou, sobre o conjunto ISCX VPN-nonVPN, a viabilidade de classificar tráfego cifrado com redes neurais profundas em tarefas de categorização do tráfego e identificação de aplicações. Seu foco, contudo, está na arquitetura de aprendizado e no uso de uma representação byte a byte dos pacotes pré-processados, e não na construção auditável e incremental da cadeia experimental, aspecto central deste trabalho.

O trabalho mais próximo do recorte adotado neste artigo é o de Kotak et al. [Kotak et al. 2025], que identifica inconsistências no ISCXVPN2016, constrói uma base de dados própria e argumenta que abordagens baseadas em fluxos tradicionais não se aplicam diretamente ao tráfego VPN real, no qual os campos de endereçamento e portas das aplicações encapsuladas não são visíveis ao observador. Além disso, os autores propõem janelas deslizantes de atributos por segundo e empregam o InceptionTime, obtendo acurácia de 93–95% para macrocategoria e 84–91% para serviço.

Akbari et al. [Akbari et al. 2021] mostram que a classificação de tráfego cifrado é sensível aos sinais preservados na representação dos dados, exigindo cuidado na escolha e engenharia dos atributos e na definição do ponto de captura. Essa observação dialoga com a opção metodológica deste trabalho de observar apenas o enlace do túnel VPN, restringindo a análise aos atributos disponíveis a um monitor intermediário.

Tabela 1. Comparação metodológica entre trabalhos relacionados e o presente artigo.

Trabalho	Base de Dados	Unidade de Amostragem	Modelo(s)	Aval. por Sessão
Draper-Gil [Draper-Gil et al. 2016]	ISCXVPN2016	Fluxos por 5-tupla	C4.5, KNN	Não
Lotfollahi [Lotfollahi et al. 2020]	ISCX VPN-nonVPN	Pacotes pré-processados	SAE, 1D-CNN	Não
Kotak [Kotak et al. 2025]	Própria	Janelas temporais	InceptionTime, LSTM	Não
Este Trabalho	**Própria**	**Janelas temporais**	**GBM, florestas, SVM e linear**	**Sim**

Em síntese, a Tabela 1 evidencia algumas das principais diferenças metodológicas entre este trabalho e os estudos anteriores. Em particular, nossa proposta adota um cenário experimental que observa apenas o enlace do túnel VPN, aproximando a coleta da perspectiva de um monitor de rede intermediário. A base de dados é composta por sessões auditadas antes do janelamento e permanece extensível a novas capturas. Além disso, o protocolo de avaliação mantém todas as janelas de uma mesma sessão na mesma partição, reduzindo o risco de vazamento entre treino e teste e favorecendo a avaliação em sessões não vistas durante o treinamento. O conjunto dessas decisões metodológicas, aliado à

avaliação de diferentes famílias de classificadores, produz F1-Macro de **98,83%** na tarefa binária e **97,15%** na tarefa multiclasse de oito serviços, evidenciando a eficácia da ferramenta proposta para classificação de tráfego cifrado em VPNs.

3. Cenário Experimental

3.1. Topologia e ponto de observação

O ambiente experimental utiliza duas máquinas virtuais Lubuntu 22.04 LTS no Oracle VirtualBox (Figura 1). O **nó monitor** (VM1) opera como servidor VPN e ponto de observação: possui uma interface NAT para acesso à Internet e uma interface de rede interna para comunicação com o nó gerador. O **nó gerador** (VM2) opera como cliente VPN e gerador de tráfego, acessando a Internet exclusivamente por meio do túnel OpenVPN (modo *tun*, UDP, AES-256-GCM, HMAC-SHA256), sem rotas alternativas de saída.

A escolha de observar o tráfego no nó monitor, no enlace do túnel, tem motivação metodológica explícita: capturar ao lado do nó gerador introduziria artefatos locais desse nó, como processos auxiliares do sistema operacional, resolução de DNS local e efeitos do navegador, que não representam o serviço sob a perspectiva de um monitor de rede intermediário [Shen et al. 2023]. Esse deslocamento do ponto de observação é análogo ao adotado em estudos de *website fingerprinting*, nos quais a posição do observador no caminho da comunicação define os sinais disponíveis para inferência [Cherubin et al. 2022].

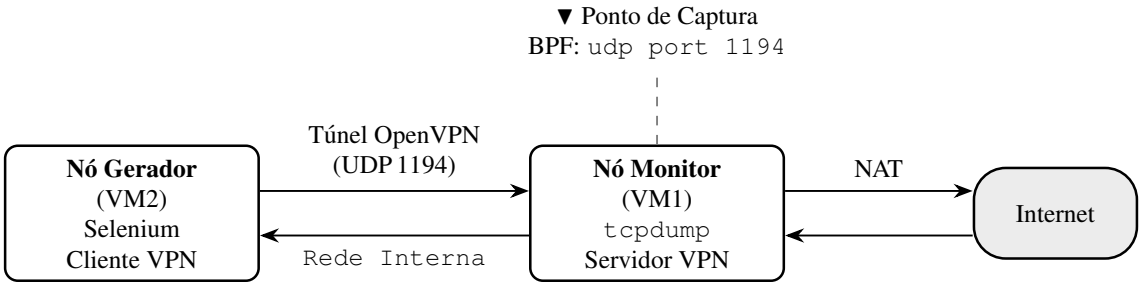

Figura 1. Topologia do laboratório virtualizado. O nó gerador acessa a Internet exclusivamente via túnel OpenVPN; o nó monitor captura o tráfego cifrado no enlace do túnel antes do encaminhamento à Internet por NAT.

3.2. Arquitetura de captura e geração de tráfego

A captura de pacotes é realizada com `tcpdump` no nó monitor, com filtro BPF restrito ao tráfego do túnel (`udp port 1194`). O processo é acionado remotamente do nó gerador via SSH em modo *batch* e encapsulado em unidades transitórias do `systemd` (`systemd-run --collect`), eliminando a dependência de sessões SSH persistentes e tornando o ciclo de captura robusto para execuções longas e automatizadas.

A geração de tráfego é realizada no nó gerador por automação em navegador real com Selenium WebDriver [Griessel et al. 2022], permitindo controlar programaticamente eventos de interação: carregamento de páginas, rolagem, cliques, reprodução de mídia e períodos de inatividade. O resultado de cada execução é um par de artefatos com nomenclatura rastreável: um arquivo PCAP com o tráfego cifrado capturado no enlace e um arquivo JSON com metadados de execução para auditoria. Os metadados de execução registram identificação rastreável da sessão, parâmetros de captura, como duração efetiva,

situação de execução e estabilidade de conexão; para sessões de transmissão de vídeo, os metadados incluem adicionalmente métricas de qualidade de reprodução obtidas por monitoramento contínuo do elemento de vídeo. Os atributos usados na modelagem são extraídos exclusivamente do PCAP; os metadados de execução servem como trilha de auditoria e não são usados como atributos nos modelos.

3.3. Seleção e caracterização dos serviços

Foram coletados oito serviços distribuídos em duas macrocategorias, com 100 sessões por serviço, mantendo o conjunto perfeitamente balanceado entre classes e evitando vieses decorrentes de classe dominante [He and Garcia 2009]. A seleção priorizou cobertura de perfis comportamentais distintos dentro de cada macrocategoria, de modo que as assinaturas de tráfego intra-classe reflitam a diversidade de uso real (Tabela 2).

Tabela 2. Composição do acervo por macrocategoria e serviço.

Macrocategoria	Serviço	Sessões	Tam. Médio	Tam. Total	Dur. Média (s)	Dur. Total (s)
web browsing	Wikipedia	100	64,2 MB	6,42 GB	318	31.830
	G1	100	141,8 MB	14,18 GB	322	32.175
	Amazon	100	101,3 MB	10,13 GB	316	31.628
	Reddit	100	119,9 MB	11,99 GB	311	31.137
streaming	YouTube	100	89,8 MB	8,98 GB	322	32.173
	YouTube Shorts	100	144,9 MB	14,49 GB	312	31.173
	Twitch	100	162,1 MB	16,21 GB	358	35.776
	Netflix	100	117,4 MB	11,74 GB	314	31.424
Total	8 serviços	**800**	117,7 MB	**94,13 GB**	322	**257.316**

Na macrocategoria de navegação web (*web browsing*), os serviços representam perfis complementares de interação: Wikipedia caracteriza navegação predominantemente textual, com baixa densidade de requisições após o carregamento inicial; G1 representa um portal jornalístico com conteúdo dinâmico, anúncios, mídia embutida e múltiplas requisições paralelas; Amazon introduz busca, navegação por catálogo, páginas de produto e carregamento intensivo de imagens; e Reddit representa navegação social com rolagem contínua, abertura de tópicos de discussão e alternância entre períodos de leitura e novas requisições.

Na macrocategoria de transmissão de vídeo (*streaming*), os serviços cobrem diferentes regimes de consumo de mídia: YouTube representa vídeo sob demanda de duração variada, com reprodução adaptativa, pré-carregamento progressivo e variações de taxa de bits (*bitrate*); YouTube Shorts representa consumo de vídeos curtos, com alternância frequente entre clipes e rajadas de pré-carregamento; Twitch representa transmissão ao vivo contínua, com fluxo bidirecional persistente, menor tolerância a variações de atraso e componente de bate-papo em tempo real; e Netflix representa reprodução sob demanda por catálogo, com fase inicial de preenchimento de buffer seguida de estabilização do fluxo em taxa de bits elevada.

O acervo totaliza 800 sessões válidas, 94 GB de tráfego capturado e 257.316 s de atividade de rede gerada.

4. Metodologia

4.1. Visão geral da ferramenta

A ferramenta proposta é estruturada como um pipeline metodológico incremental, organizado em seis etapas modulares e reproduzíveis (Figura 2): (1) geração automatizada de sessões de captura, produzindo pares PCAP/JSON; (2) auditoria e saneamento das sessões com base nos metadados JSON; (3) processamento do PCAP, com isolamento dos pacotes do túnel e identificação de direção, tempo e tamanho; (4) janelamento e engenharia de atributos, formando a base de dados com 80 atributos por janela; (5) divisão agrupada por identificador de sessão (`session_id`), de modo que janelas da mesma sessão permaneçam na mesma partição; e (6) classificação e avaliação dos modelos em nível de janela e de sessão. As etapas são incrementais: novas sessões são processadas e acrescentadas à base de dados sem reconstruir o conjunto já existente.

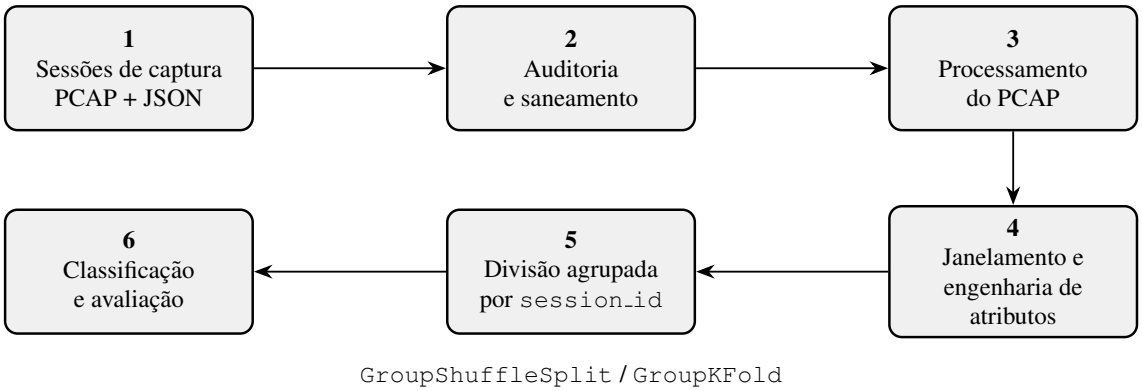

Figura 2. Pipeline metodológico da ferramenta proposta, organizado da geração das capturas à avaliação dos modelos.

4.2. Variabilidade comportamental

Um desafio reconhecido em coleta automatizada de tráfego (etapa 1) é a geração de sessões com assinaturas artificialmente uniformes, resultantes de rotinas (*scripts*) determinísticas que não reproduzem a variabilidade de interação real [Griessel et al. 2022]. Para mitigar esse risco, a variabilidade foi introduzida de forma estruturada em cada um dos oito serviços coletados.

O mecanismo central é a definição de perfis de ritmo de interação (passivo, padrão, intenso), cada qual parametrizando tempos de leitura, intensidade e frequência de rolagem, probabilidade de cliques e duração de períodos de inatividade. Esses perfis são ativados de forma probabilística e configurável, com transições a cada intervalo $\Delta t \sim \mathcal{U}(25, 70)$ s durante a sessão. Além disso, a duração-alvo inclui variação aleatória uniforme de $\pm 20\%$, evitando periodicidade artificial no acervo.

Para os serviços de navegação web (*web browsing*), um segundo eixo atua sobre a trajetória de navegação: trilhas estocásticas sem URLs fixas na Wikipedia; exploração dinâmica de notícias no G1; alternância de postagens, comentários e páginas no Reddit; e diferentes modos de navegação estratégica na Amazon.

Os serviços de transmissão de vídeo (*streaming*) introduzem variabilidade de natureza distinta: o YouTube opera sob distintos mecanismos de escolha de vídeo e resolução;

o YouTube Shorts alterna entre perfis de consumo com diferentes intensidades de rolagem e permanência; na Twitch, o canal é selecionado de um pool com prioridade de canais ao vivo; e, na Netflix, a seleção de conteúdo alterna estocasticamente entre filmes e séries.

A combinação desses eixos produz um espaço de configurações praticamente ilimitado, em que sessões de mesma classe são comportamentalmente heterogêneas por construção. A especificação completa desses parâmetros está documentada nos artefatos de reprodução disponíveis publicamente [Basaldella et al. 2026].

4.3. Auditoria e saneamento das sessões

Antes da construção da base de dados, todas as sessões são submetidas a auditoria com critérios formalizados (etapa 2). A auditoria opera ao nível de sessão: uma sessão inválida é descartada integralmente antes do janelamento, impedindo que janelas derivadas de capturas problemáticas entrem na base de dados. Essa decisão é metodologicamente relevante porque a sessão também é a unidade de agrupamento no protocolo de avaliação sem vazamento (Seção 4.6).

Os metadados JSON gerados em cada sessão operacionalizam essa auditoria. A partir deles, verificam-se critérios como duração efetiva, tamanho do PCAP, ocorrência de navegação nula, problemas de sincronização entre coleta e captura, quedas de conexão durante a execução e métricas de qualidade de reprodução. Esses metadados compõem a trilha de auditoria da sessão e não são usados como atributos dos modelos.

4.4. Janelamento e engenharia de atributos

Após a auditoria das sessões, cada PCAP válido é processado com a biblioteca `dpkt`, extraindo exclusivamente pacotes UDP pertencentes ao túnel OpenVPN (etapa 3). A direção de cada pacote é inferida pelo endereço IP local do nó monitor: pacotes destinados a esse endereço são classificados como *upstream* (cliente $\rightarrow$ servidor) e pacotes originados desse endereço como *downstream* (servidor $\rightarrow$ cliente). O intervalo entre chegadas (IAT) de cada pacote é definido como $\tau_k = t_k - t_{k-1}$, onde t_k é a marca temporal registrada pelo `tcpdump` no nó monitor, relativizada ao início da sessão.

A sessão é segmentada em janelas temporais fixas sem sobreposição, com passo (*stride*) igual a w:

$$W_j^{(i)} = \left\{ (t_k,\, \ell_k,\, d_k) : j \cdot w \leq t_k < (j+1) \cdot w \right\} \tag{1}$$

onde i identifica a sessão, j é o índice da janela dentro da sessão, ℓ_k é o tamanho do k-ésimo pacote em bytes e $d_k \in \{\textit{upstream}, \textit{downstream}\}$ indica sua direção.

Para cada janela $W_j^{(i)}$, extrai-se um vetor de atributos $\mathbf{x}_{ij} \in \mathbb{R}^{80}$ (etapa 4). Os rótulos de classe da janela (macrocategoria e serviço) são herdados da sessão de origem. Cada janela também mantém o `session_id`, utilizado no protocolo agrupado de avaliação para evitar vazamento entre treino e teste. O índice da janela, seus limites temporais e o `session_id` são preservados para rastreabilidade e particionamento, mas não são usados como atributos preditivos.

Os 80 atributos extraídos foram definidos para cobrir as principais dimensões observáveis do tráfego encapsulado sem acesso à carga útil dos pacotes (Tabela 3): volume,

taxa e direcionalidade (G1); distribuição de tamanhos de pacote e de IATs, com medidas de tendência central, dispersão, quantis e assimetria, tanto globalmente (G2, G3) quanto separados por direção (G4, G5); dinâmica temporal e estrutura de rajadas (G6, G7); e medidas de regularidade, entropia e concentração (G8–G10). A decomposição por direção (G4, G5) é particularmente relevante: ela preserva a assimetria *upstream*/*downstream*, que varia sistematicamente entre serviços, com transmissão de vídeo concentrando volume no sentido *downstream* e navegação web exibindo alternância mais equilibrada entre os dois sentidos.

Tabela 3. Grupos de atributos extraídos por janela temporal.

Grupo	Descrição	#
G1	Volume, taxa e direcionalidade	12
G2	Estatísticas globais de tamanho de pacote	11
G3	Estatísticas globais de IAT	12
G4	Tamanho de pacote por direção (*upstream*/*downstream*)	6
G5	IAT por direção (*upstream*/*downstream*)	6
G6	Análise de rajadas (*burst*)	13
G7	Distribuição temporal da carga intra-janela	8
G8	Padrão de atividade e ociosidade	4
G9	Entropia e regularidade temporal	6
G10	Concentração de bytes e tamanhos	2
Total		**80**

Algumas medidas sintetizam propriedades centrais das janelas e ajudam a capturar diferenças empiricamente observadas entre os serviços. O coeficiente de variação dos intervalos entre chegadas, $\mathrm{CV}(\tau) = \sigma_\tau / \mu_\tau$, mede a irregularidade temporal do fluxo: valores maiores indicam chegadas mais dispersas, enquanto valores menores indicam ritmo mais uniforme. A entropia de Shannon,

$$H(X) = -\sum_{b=1}^{B} \hat{p}_b \log_2 \hat{p}_b, \tag{2}$$

onde B é o número de intervalos do histograma e $\hat{p}_b$ a fração de pacotes em cada intervalo, quantifica a dispersão da distribuição de tamanhos de pacote ou de IATs dentro da janela. O coeficiente de Gini [Hurley and Rickard 2009],

$$G = \frac{2\sum_{k=1}^{n} k\, x_{(k)}}{n \sum_{k=1}^{n} x_{(k)}} - \frac{n+1}{n}, \tag{3}$$

onde $x_{(k)}$ é o k-ésimo tamanho de pacote em ordem crescente e n o total de pacotes na janela, mede a concentração da distribuição de tamanhos: valores maiores indicam maior desigualdade entre os tamanhos observados. Por fim, a análise de rajadas quantifica sequências consecutivas de pacotes na mesma direção, capturando alternância direcional e volume por rajada.

4.5. Seleção do tamanho de janela

A escolha de w afeta diretamente o compromisso entre desempenho preditivo, resolução temporal e custo computacional. Janelas muito curtas aumentam a massa amostral, mas podem capturar apenas fragmentos sem estrutura discriminativa; janelas muito longas reduzem o número de amostras e aumentam a latência de decisão em uso operacional [Kotak et al. 2025]. Para formalizar essa escolha, este trabalho adota a seguinte função de utilidade:

$$U(w) = F_1^{\text{sess}}(w) \; - \; \lambda \cdot \frac{w}{w_0} \; - \; \mu \cdot \frac{w_0}{w} \tag{4}$$

onde $F_1^{\text{sess}}(w)$ é o F1-Macro avaliado em nível de sessão para janela de tamanho w, $w_0 = 10$ s é o tamanho de referência para normalização, λ penaliza janelas longas (latência de decisão) e μ penaliza janelas curtas (custo de amostragem). Os valores $\lambda = \mu = 0{,}02$ foram calibrados para manter penalização moderada no intervalo avaliado, preservando o papel dominante do desempenho preditivo na seleção.

Para a análise de sensibilidade, foram construídas três bases de dados, uma para cada tamanho de janela ($w \in \{5, 10, 20\}$ s). O valor $w = 10$ s maximizou $U(w)$ de forma consistente em ambas as tarefas no modelo de melhor desempenho, sendo adotado como tamanho de janela padrão da ferramenta. Os resultados numéricos detalhados são apresentados na Seção 5.

4.6. Protocolo de avaliação sem vazamento

Em bases de dados construídas por janelamento de sessões longas, janelas da mesma sessão compartilham informação contextual, como padrão médio de taxa de bits, qualidade de rede no momento da coleta e condições específicas da execução. Dividir essas janelas entre treino e teste constitui uma forma de vazamento de dados, capaz de inflar artificialmente as métricas [Kaufman et al. 2012, Zhao et al. 2025].

Para controlar esse risco, dois protocolos são executados em paralelo (etapa 5). O protocolo de referência usa divisão aleatória estratificada, com 20% dos dados reservados para o conjunto de teste (*holdout*) e validação cruzada de cinco partições sobre o conjunto de treino, sem restringir a separação por sessão. O protocolo agrupado usa `GroupShuffleSplit` e `GroupKFold` com agrupamento por `session_id`, garantindo que todas as janelas de uma mesma sessão estejam inteiramente em uma única partição (treino ou teste). O delta agrupado (*grouped delta*), definido como a diferença entre os dois protocolos, serve como diagnóstico de vazamento: valores próximos de zero sugerem ausência de inflação relevante no protocolo de referência.

Ambos os protocolos reportam métricas em dois níveis. Em nível de janela, as métricas são F1-Macro e acurácia balanceada. Em nível de sessão, as predições são agregadas pela média das probabilidades estimadas:

$$\hat{y}_i = \arg\max_c \; \frac{1}{|J_i|} \sum_{j \in J_i} \hat{p}_{ij}^{(c)} \tag{5}$$

onde $J_i = \{j \, : \, W_j^{(i)} \in \text{conjunto de teste}\}$ é o conjunto de janelas da sessão i

alocadas ao teste e $\hat{p}_{ij}^{(c)}$ é a probabilidade estimada pelo modelo para a classe c na janela j. A avaliação em nível de sessão é mais conservadora e mais próxima do cenário operacional, em que a decisão é tipicamente tomada por sessão completa. Esse protocolo é relevante porque avalia a capacidade dos modelos de generalizar para novas sessões, em vez de apenas reconhecer padrões específicos de capturas já presentes no treinamento.

4.7. Classificadores e configuração experimental

Seis classificadores foram avaliados nas tarefas de macrocategoria e serviço (etapa 6): LightGBM [Ke et al. 2017], XGBoost [Chen and Guestrin 2016], Random Forest [Breiman 2001], Extra Trees, SVM com kernel RBF [Cortes and Vapnik 1995] e Regressão Logística [Pedregosa et al. 2011], além de um classificador *Dummy* como limite inferior de referência. Todos foram avaliados com configurações fixas, sem busca de hiperparâmetros e com semente aleatória fixa ($s = 42$), caracterizando uma linha de base controlada. O conjunto cobre métodos baseados em árvores com reforço de gradiente, comitês de árvores, métodos de margem e modelos lineares, permitindo comparar diferentes famílias de classificadores.

5. Resultados

5.1. Caracterização da base de dados

A Tabela 4 apresenta as estatísticas da base de dados principal ($w = 10$ s) por serviço. Em média, cada janela contém 3.641 pacotes e 3,66 MB de tráfego. Serviços de transmissão de vídeo (*streaming*) com maior densidade de mídia (Twitch, YouTube Shorts) exibem volume consideravelmente maior do que serviços de navegação de menor volume (Wikipedia), evidenciando perfis de tráfego distintos.

Tabela 4. Estatísticas da base de dados principal ($w = 10$ s) por serviço. Pkts/jan.: pacotes por janela (média); Vol./jan.: volume médio por janela (MB).

Categoria	Serviço	Jan.	Jan./Sess.	Pkts/Jan.	Vol./Jan. (MB)
web browsing	Amazon	3.198	32,0	3.312	3,12
	G1	3.250	32,5	4.898	4,28
	Reddit	3.150	31,5	3.892	3,74
	Wikipedia	3.163	31,6	1.844	2,00
streaming	Netflix	3.149	31,5	3.430	3,67
	Twitch	3.054	30,5	4.796	5,23
	YouTube	3.228	32,3	2.484	2,74
	YouTube Shorts	3.133	31,3	4.515	4,55
Total/Média	—	**25.325**	**31,7**	**3.641**	**3,66**

5.2. Tarefa de macrocategoria (binária)

A Tabela 5 apresenta os resultados em nível de janela na tarefa binária de classificação de macrocategoria. O LightGBM lidera com F1-Macro de 0,9883 no conjunto de teste (holdout). A diferença entre o desempenho no conjunto de teste e na validação cruzada (Δ_{CV}) é de apenas $+0{,}0008$, indicando generalização estável e ausência de sobreajuste. A avaliação com protocolo agrupado por sessão resultou em delta agrupado de $+0{,}0017$, indicando que a separação aleatória não produziu inflação mensurável por vazamento entre janelas da mesma sessão.

Tabela 5. Resultados na tarefa binária de macrocategoria (*web browsing* vs. *streaming*). F1-CV: F1-Macro médio na validação cruzada; Δ_{CV}: diferença entre F1-Macro no teste e F1-CV.

Modelo	F1-Macro	F1-CV	Δ_{CV}	Bal. Acc	AUC
LightGBM	**0,9883**	0,9875	+0,0008	0,9883	0,9991
XGBoost	0,9866	0,9865	+0,0001	0,9865	0,9992
Random Forest	0,9836	0,9824	+0,0012	0,9836	0,9981
Extra Trees	0,9834	0,9830	+0,0004	0,9834	0,9979
SVM (RBF)	0,9668	0,9718	−0,0050	0,9668	0,9943
Reg. Logística	0,9234	0,9357	−0,0123	0,9235	0,9798
Dummy	0,3350	0,3351	−0,0001	0,5000	0,5000

O classificador *Dummy* (classe majoritária *web browsing*) obtém F1-Macro de 0,3350 e acurácia balanceada de 0,50; o LightGBM supera esse piso em $+0{,}6533$, confirmando que os atributos preservam informação discriminativa sem acesso à carga útil dos pacotes. A diferença de $+0{,}0649$ entre Regressão Logística e LightGBM indica ganho de modelos não lineares na separação entre as macrocategorias.

5.3. Tarefa de serviço (multiclasse)

A Tabela 6 apresenta os resultados em nível de janela na tarefa de classificação do serviço de origem do tráfego entre oito classes. O LightGBM mantém a liderança com F1-Macro de 0,9715 no conjunto de teste. Alinhado à tarefa binária, o Δ_{CV} de $+0{,}0052$ indica generalização estável e ausência de sobreajuste. A avaliação com protocolo agrupado por sessão resultou em delta agrupado de $-0{,}0019$, indicando ausência de inflação mensurável por vazamento entre janelas da mesma sessão. O Top-3 de 0,9968 revela que, nos raros casos de erro, a classe correta quase sempre aparece entre as três maiores probabilidades estimadas pelo modelo.

Tabela 6. Resultados na tarefa multiclasse de serviço (8 classes). F1-CV: F1-Macro médio na validação cruzada; Δ_{CV}: diferença entre F1-Macro no teste e F1-CV.

Modelo	F1-Macro	F1-CV	Δ_{CV}	Bal. Acc	Top-3
LightGBM	**0,9715**	0,9663	+0,0052	0,9716	0,9968
XGBoost	0,9692	0,9645	+0,0047	0,9693	0,9970
Random Forest	0,9615	0,9539	+0,0076	0,9616	0,9941
Extra Trees	0,9603	0,9522	+0,0081	0,9604	0,9931
SVM (RBF)	0,9332	0,9276	+0,0056	0,9336	0,9878
Reg. Logística	0,8991	0,8890	+0,0101	0,8999	0,9803
Dummy	0,0284	0,0284	0,0000	0,1250	0,3795

O classificador *Dummy* (classe majoritária G1) obtém F1-Macro de 0,0284 e acurácia balanceada de 0,125; o LightGBM supera esse piso em $+0{,}9431$, confirmando o poder discriminativo dos atributos mesmo entre oito serviços. A diferença de $+0{,}0724$ entre Regressão Logística e LightGBM, superior à da tarefa binária ($+0{,}0649$), indica ganho de modelos não lineares na separação entre serviços de mesma macrocategoria. O SVM (RBF), a 0,0383 abaixo do LightGBM, sugere que métodos baseados em árvores de gradiente capturam melhor as interações entre atributos de volume, IAT e direcionalidade.

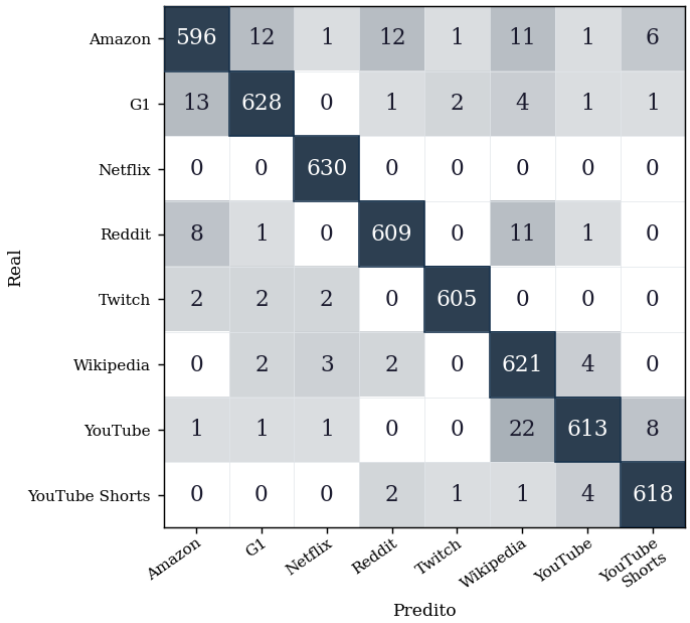

Figura 3. Matriz de confusão do LightGBM na tarefa de serviço.

A Figura 3 complementa essa análise ao localizar os erros em nível de janela. As confusões são pouco numerosas e concentram-se em serviços com padrões temporais parcialmente semelhantes, como YouTube e YouTube Shorts, além de casos pontuais envolvendo YouTube e Wikipedia. A Amazon apresenta erros mais dispersos entre os serviços de navegação, compatíveis com a heterogeneidade de suas páginas de busca, catálogo e produto. Em contraste, todas as 630 janelas da Netflix foram reconhecidas, resultado possivelmente favorecido pelo protocolo padronizado de reprodução contínua. Esse desempenho deve ser interpretado no contexto controlado do experimento, e não como uma assinatura universal do serviço.

5.4. Sensibilidade ao tamanho de janela

Tabela 7. Sensibilidade ao tamanho de janela w em nível de sessão.

w (s)	Janelas	Macrocategoria		Serviço	
		F_1^{sess}	$U(w)$	F_1^{sess}	$U(w)$
5	50.252	1,0000	0,9500	0,9690	0,9190
10	**25.325**	**1,0000**	**0,9600**	**0,9883**	**0,9483**
20	12.872	1,0000	0,9500	0,9883	0,9383

A análise de sensibilidade com $w \in \{5, 10, 20\}$ s confirmou $w = 10$ s como melhor compromisso em ambas as tarefas (Tabela 7). Com $w = 5$ s, o maior número de janelas não compensou a queda de F_1^{sess} na tarefa de serviço ($0{,}9690$ vs. $0{,}9883$) nem o custo de amostragem, resultando em $U(5) = 0{,}9190$. Com $w = 20$ s, o desempenho de sessão empatou com $w = 10$ s na tarefa de serviço, mas a penalidade de latência reduziu $U(20)$ a $0{,}9383$. Na tarefa de macrocategoria, todos os tamanhos atingiram $F_1^{\text{sess}} = 1{,}00$, tornando $U(w)$ o critério decisivo.

5.5. Atributos mais relevantes

A inspeção dos atributos mais utilizados pelo LightGBM na tarefa de serviço, medida pela contagem de divisões (*split count*), isto é, pelo número de vezes em que cada atributo foi selecionado para particionar um nó nas árvores do modelo, indica quais grupos de atributos mais sustentam a discriminação multiclasse (Figura 4). Os 20 atributos mais frequentes pertencem quase exclusivamente aos grupos G3 (IAT global) e G5 (IAT por direção), que juntos ocupam 15 das 20 posições; `iat_min` lidera com $11{,}4\%$ das divisões.

Esse padrão reforça o papel central do ritmo de chegada dos pacotes: influenciado pela interação e pela entrega de conteúdo de cada serviço, o IAT preserva informação discriminativa mesmo após o encapsulamento VPN. Atributos de tamanho por direção (G4) e de entropia/regularidade (G9) complementam a ordenação, indicando papel secundário, mas relevante, da distribuição de tamanhos na discriminação entre serviços.

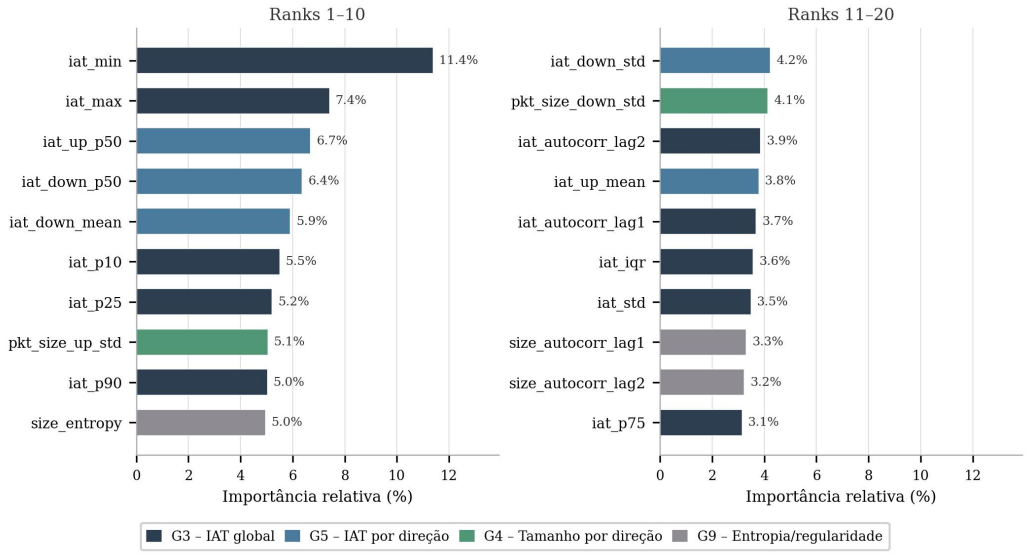

Figura 4. Top-20 atributos por *split count* no LightGBM (tarefa de serviço).

6. Conclusão

Este trabalho apresentou uma ferramenta reproduzível para classificar tráfego cifrado em VPNs sem inspeção da carga útil. Suas contribuições são: (i) metodologia de coleta que separa os nós gerador e monitor, produzindo atributos da perspectiva de um observador intermediário; (ii) protocolo de variabilidade comportamental com automação Selenium, navegação estocástica, ritmo probabilístico e parâmetros configuráveis por serviço; e (iii) pipeline incremental com auditoria formal por sessão, extração de atributos e avaliação com controle explícito de vazamento por `session_id`.

O acervo resultante compreende 800 sessões auditadas cobrindo oito serviços em duas macrocategorias, com 25.325 janelas de 10 s balanceadas entre os serviços. Os 80 atributos estatísticos extraídos sem acesso à carga útil dos pacotes (cobrindo volume, IAT, direcionalidade, regularidade, entropia e concentração de carga) revelaram alto poder discriminativo. Na tarefa de classificação de macrocategoria, o LightGBM alcançou F1-Macro de $0{,}9883$ com delta agrupado de apenas $+0{,}0017$, indicando ausência de inflação

mensurável por vazamento de dados entre janelas da mesma sessão. Na tarefa multiclasse de serviço, o mesmo modelo obteve F1-Macro de $0{,}9715$, com delta agrupado de $-0{,}0019$. A separação entre macrocategorias mostrou-se elevada, indicando que os perfis de transmissão de vídeo e navegação web permanecem distintos no nível do túnel VPN. As confusões remanescentes ocorrem em baixa frequência e concentram-se em serviços com padrões temporais parcialmente sobrepostos, o que é esperado em uma classificação baseada apenas em atributos estatísticos do túnel cifrado. A análise de sensibilidade ao tamanho de janela, formalizada pela função $U(w)$, confirmou $w = 10$ s como ponto de equilíbrio entre desempenho preditivo, resolução temporal e custo amostral.

Esses resultados demonstram que o tráfego encapsulado em túnel VPN preserva padrões estatísticos suficientes para identificar com alta confiança tanto a macrocategoria quanto o serviço específico, sem qualquer acesso à carga útil dos pacotes. As aplicações práticas vão muito além da segurança: na gestão de redes, operadoras e administradores podem aplicar políticas diferenciadas de qualidade de serviço (QoS) com base no tipo de tráfego detectado, priorizando, por exemplo, fluxos sensíveis a atraso em momentos de congestionamento, ou alocando largura de banda de forma proporcional à demanda por serviço [Nguyen and Armitage 2008]. Em redes corporativas, a identificação do serviço viabiliza conformidade de uso (*compliance*), detecção de acesso a plataformas não autorizadas e auditoria de tráfego sem inspeção de conteúdo. Do ponto de vista de negócio, provedores de Internet podem refinar modelos de tarifação, planejamento de capacidade e estratégias de roteamento com base em perfis de consumo identificados automaticamente [Shen et al. 2023]. Esse conjunto de aplicações reforça a relevância operacional da ferramenta em cenários nos quais a inspeção da carga útil não é viável ou desejável.

Como limitações, o experimento foi conduzido em ambiente de laboratório com conectividade estável e uso de um único protocolo VPN (OpenVPN), sem interferência de múltiplos usuários e com sessões de qualidade controlada. As sessões foram geradas por automação Selenium com protocolo de variabilidade comportamental explícita, mas uma validação complementar com sessões manuais não foi realizada [Griessel et al. 2022]. Os classificadores foram avaliados com configurações fixas, sem busca de hiperparâmetros, caracterizando uma linha de base controlada; ajustes específicos podem alterar o desempenho observado. A observação no enlace do túnel adota a perspectiva de um monitor intermediário; outros pontos de captura podem revelar atributos distintos [Akbari et al. 2021].

6.1. Trabalhos Futuros

Trabalhos futuros podem explorar: (i) a expansão do acervo para novos serviços (redes sociais, videoconferência, jogos online e VoIP), mantendo o rigor da coleta reproduzível e da auditoria por sessão para avaliar a generalização em espaços de classes mais amplos e heterogêneos; (ii) um módulo de classificação em tempo real, capaz de processar janelas, produzir predições e acionar respostas automáticas, como marcação DSCP, ajuste de encaminhamento ou alerta de conformidade; (iii) a generalização entre bases distintas e a comparação com abordagens recentes de aprendizado profundo aplicadas a séries temporais [Kotak et al. 2025]; e (iv) diferentes protocolos VPN e ambientes operacionais, com múltiplos usuários e degradação controlada da rede. Para favorecer a reprodução e a extensão do estudo, o código-fonte, a documentação e os conjuntos de atributos estão disponíveis publicamente [Basaldella et al. 2026].

Agradecimentos

Este trabalho é parcialmente financiado pelo CNPq (408255/2023-4 e 407304/2025-8), CAPES (88887.987121/2024-00 e 88881.987122/2024-01), COFECUB (Ma1074/25), FAPERJ e RNP e faz parte do INCT de Redes de Comunicação e Internet das Coisas Inteligentes (ICoNIoT), financiado pelo CNPq (405940/2022-0) e pela CAPES (88887.954253/2024-00).

Referências

Akbari, I., Salahuddin, M. A., Ven, L., Limam, N., Boutaba, R., Mathieu, B., Moteau, S., and Tuffin, S. (2021). A look behind the curtain: Traffic classification in an increasingly encrypted web. In *Abstract Proceedings of the 2021 ACM SIGMETRICS / International Conference on Measurement and Modeling of Computer Systems*, pages 23–24. Association for Computing Machinery.

Anderson, B. and McGrew, D. (2017). Machine learning for encrypted malware traffic classification: Accounting for noisy labels and non-stationarity. In *Proceedings of the 23rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 1723–1732. Association for Computing Machinery.

Basaldella, E., Mascarenhas, D. M., and Moraes, I. M. (2026). VPN encrypted traffic classification tool. Repositório no GitHub: `https://github.com/enzobasaldella/vpn-encrypted-traffic-classification-sbseg-2026`. Artefatos de reprodução do SBSeg 2026. Acesso em: 3 ago. 2026.

Breiman, L. (2001). Random forests. *Machine Learning*, 45(1):5–32.

Chen, T. and Guestrin, C. (2016). XGBoost: A scalable tree boosting system. In *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 785–794.

Cherubin, G., Jansen, R., and Troncoso, C. (2022). Online website fingerprinting: Evaluating website fingerprinting attacks on Tor in the real world. In *31st USENIX Security Symposium (USENIX Security 22)*, pages 753–770.

Cortes, C. and Vapnik, V. (1995). Support-vector networks. *Machine Learning*, 20(3):273–297.

Draper-Gil, G., Lashkari, A. H., Mamun, M. S. I., and Ghorbani, A. A. (2016). Characterization of encrypted and VPN traffic using time-related features. In *Proceedings of the 2nd International Conference on Information Systems Security and Privacy (ICISSP)*, pages 407–414.

Griessel, A., Stephan, M., Mieth, M., Kellerer, W., and Krämer, P. (2022). RLBrowse: Generating realistic packet traces with reinforcement learning. In *2022 IEEE/IFIP Network Operations and Management Symposium (NOMS)*, pages 1–7.

He, H. and Garcia, E. A. (2009). Learning from imbalanced data. *IEEE Transactions on Knowledge and Data Engineering*, 21(9):1263–1284.

Hurley, N. and Rickard, S. (2009). Comparing measures of sparsity. *IEEE Transactions on Information Theory*, 55(10):4723–4741.

Kaufman, S., Rosset, S., Perlich, C., and Stitelman, O. (2012). Leakage in data mining: Formulation, detection, and avoidance. *ACM Transactions on Knowledge Discovery from Data*, 6(4):15:1–15:21.

Ke, G., Meng, Q., Finley, T., Wang, T., Chen, W., Ma, W., Ye, Q., and Liu, T.-Y. (2017). LightGBM: A highly efficient gradient boosting decision tree. In *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc.

Kotak, J., Yankelev, I., Bibi, I., Elovici, Y., and Shabtai, A. (2025). VPN-encrypted network traffic classification using a time-series approach. *IEEE Transactions on Network and Service Management*, 22(2):2225–2242.

Lotfollahi, M., Jafari Siavoshani, M., Shirali Hossein Zade, R., and Saberian, M. (2020). Deep Packet: A novel approach for encrypted traffic classification using deep learning. *Soft Computing*, 24(3):1999–2012.

Nguyen, T. T. T. and Armitage, G. (2008). A survey of techniques for internet traffic classification using machine learning. *IEEE Communications Surveys & Tutorials*, 10(4):56–76.

Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., Vanderplas, J., Passos, A., Cournapeau, D., Brucher, M., Perrot, M., and Duchesnay, É. (2011). Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 12:2825–2830.

Razooqi, Y. S. and Pekár, A. (2025). VPN traffic analysis: A survey on detection and application identification. *IEEE Access*, 13:132830–132848.

Shen, M., Ye, K., Liu, X., Zhu, L., Kang, J., Yu, S., Li, Q., and Xu, K. (2023). Machine learning-powered encrypted network traffic analysis: A comprehensive survey. *IEEE Communications Surveys & Tutorials*, 25(1):791–824.

Zhao, Y., Dettori, G., Boffa, M., Vassio, L., and Mellia, M. (2025). The sweet danger of sugar: Debunking representation learning for encrypted traffic classification. In *Proceedings of the ACM SIGCOMM 2025 Conference*, pages 296–310. Association for Computing Machinery.