探索密流分析鲜度模型：从“数据留存”到“价值沉淀”

卞华峰[1,2],郑纪成[1,2],马小博[1,2,3*],刘宏桃[1,2],管晓宏[1,2]

(1. 西安交通大学电子与信息学部,陕西西安 710049;2. 智能网络与网络安全教育部重点实验室,陕西西安 710049;
3. 陕西省计算机网络重点实验室,陕西西安 710049)

摘 要： 众所周知,网络流量呈海量化与密态化发展趋势。受限于计算与存储资源,流量分析系统难以对高速海量密流进行在线建模并全量存储,通常采用“先存后筛”的序贯策略,即先完整留存目标密流后,再筛选代表性样本开展分析。此策略依赖人工定义特定留存目标(如特定IP地址范围),难以应对流量的动态演化特性,导致留存数据的边际价值不断递减。即使人为提升留存目标多样性,由于缺乏系统化价值评价体系,不同留存数据仍难以实现价值沉淀,无法反向优化数据留存策略。为此,本文提出一种密流分析鲜度方法(FreshHunter)。该方法从时间、空间与语义三个维度系统化建模密流鲜度,时间维度采用牛顿水冷定律刻画密流的自然老化特征,空间和语义维度提取跨类型、跨行为的通用个体结构特征。进一步地,FreshHunter构建了鲜度感知的对比学习框架,通过引入可学习类中心,增强密流特征嵌入与类中心间的一致性,确保密流样本群体多样性,并实现“边筛边存”的高鲜度密流样本在线留存机制。在留存过程中,FreshHunter将高鲜度密流的特征嵌入不断聚合至可学习类中心,以构建代表性密流特征的统一表示;同时结合特征漂移检测与自适应校正机制,动态更新类中心,以捕捉密流特征的时序性演化。由此,FreshHunter同时捕捉高价值、代表性的密流样本及其演化,促成了从瞬时数据捕获到长期知识累积的升级,为下游密流分析提供前向知识沉淀、后向持续优化的数据支撑。本文在真实数据集上开展广泛评估,覆盖物联网(Internet of Things,IoT)终端、手机App和网站识别三类典型密流分析任务及九种主流分析算法。实验结果表明,FreshHunter可提升下游密流分析算法的F_1分数约15%~60%,优于Random、动态时间规整(Dynamic Time Warping,DTW)、树模型沙普利加性解释(Tree SHAPley additive explanations,TreeSHAP)三种基线密流筛选方法。在特征漂移场景下,FreshHunter仍保持较强适应性,提升F_1分数约20%。本研究为资源受限条件下密流分析的数据利用率和模型性能增强提供了新的技术路径。

关键词： 加密流量;流量指纹识别;流量价值;概念漂移;机器学习

基金项目： 国家自然科学基金(No.U23A20332,No.62272381);陕西省杰出青年科学基金(No.2023-JC-JQ-50)

中图分类号： TP393.08 **文献标识码：** A **文章编号：** 0372-2112(2026)04-1446-14

电子学报URL:http://www.ejournal.org.cn **DOI:**10.12263/DZXB.20251008

Exploring Freshness Modeling for Encrypted Traffic Analysis: From “Data Retention” to “Value Accumulation”

BIAN Huafeng[1,2], ZHENG Jicheng[1,2], MA Xiaobo[1,2,3*], LIU Hongtao[1,2], GUAN Xiaohong[1,2]

(1. *Faculty of Electronic and Information Engineering*, *Xi'an Jiaotong University*, *Xi'an*, *Shaanxi* 710049, *China*;
2. *Ministry of Education Laboratory for Intelligent Networks and Network Security*, *Xi'an*, *Shaanxi* 710049, *China*;
3. *Shaanxi Provincial Key Laboratory of Computer Networks*, *Xi'an*, *Shaanxi* 710049, *China*)

Abstract: As is well known, network traffic has become increasingly massive and encrypted. Due to limitations in computational and storage resources, traffic analysis systems struggle to perform online modeling and full-volume storage of high-speed encrypted traffic. Existing systems typically adopt a “store-then-select” strategy, wherein all target encrypted traffic is fully retained before being selected for analysis. However, this strategy relies on manually defined retention targets (e.g., specific IP ranges), making it difficult to adapt to dynamic traffic evolution and resulting in a diminishing marginal value of retained data. Despite deliberate efforts to increase the diversity of retention targets, the absence of a systematic value evaluation framework hinders the accumulation of data value and precludes the feedback-driven optimization of retention strategies. To address these challenges, this paper proposes FreshHunter, a freshness modeling approach for encrypted traffic analysis. FreshHunter systematically models the freshness of encrypted traffic from three dimensions: temporal, spatial, and semantic. The temporal dimension adopts Newton's law of cooling to describe the natural aging of encrypted traf-

收稿日期:2026-01-08;录用日期:2026-02-09;责任编委:沈蒙

*通信作者:马小博

fic, whereas the spatial and semantic dimensions extract structural features that are universal across types and behaviors. In addition, FreshHunter constructs a freshness-aware contrastive learning framework that incorporates learnable class centers to enhance the consistency between traffic embeddings and their corresponding centers, preserve the diversity of traffic samples, and thereby enable a “select-while-storing” mechanism for retaining high-freshness samples. During retention, FreshHunter continuously aggregates the embeddings of high-freshness encrypted traffic into learnable class centers, thereby forming a unified representation of representative traffic features. Meanwhile, it employs feature drift detection and adaptive correction mechanisms to dynamically update these class centers and capture the temporal evolution of traffic features. Thus, FreshHunter captures high-value, representative encrypted traffic samples along with their temporal evolution, facilitating an upgrade from instantaneous data capture to sustained knowledge accumulation, while supporting forward knowledge consolidation and backward adaptive optimization for downstream encrypted traffic analysis. Comprehensive evaluations are conducted on real-world datasets covering three representative encrypted traffic analysis tasks (i.e., internet of things (IoT) device, mobile App, and website identification) across nine mainstream analysis algorithms. Experimental results show that FreshHunter improves the F_1-score of downstream encrypted traffic analysis algorithms by approximately 15% to 60%, outperforming the three baseline traffic selection methods: Random, dynamic time warping (DTW), and tree shapley additive explanations (TreeSHAP). Even under traffic feature drift scenarios, FreshHunter maintains strong adaptability, improving algorithm performance by around 20%. Overall, this study offers a novel technical pathway for advancing data efficiency and model adaptability in encrypted traffic analysis under resource-constrained conditions.

Keywords: encrypted traffic; traffic fingerprinting; traffic value; concept drift; machine learning

Foundation Item(s): National Natural Science Foundation of China (No.U23A20332, No.62272381); Shaanxi Provincial Outstanding Young Scientists Fund (No.2023-JC-JQ-50)

0　引言

智能终端与云服务的普及显著扩大了网络的攻击面,使得网络空间面临前所未有的安全威胁[1-3]。例如,缺乏完善防护机制的物联网(Internet of Things, IoT)设备尤为脆弱,易遭受远程入侵并被劫持为僵尸网络节点[1]。又如,用户在日常网络交互时,可能被诱导访问恶意应用(App)或网站(Website)[2],导致敏感信息泄露,甚至造成经济损失。为应对上述安全威胁,网络管理员通常在网关处对进出流量进行收集与分析,以便及时识别潜在的恶意行为,并实施针对性安全防护措施。

然而,在实际环境中,网络流量呈现海量化与密态化发展趋势。受限于计算与存储能力,部署在边缘网关或检测节点的流量分析系统难以对海量高速密流进行在线建模并全量存储。值得注意的是,并非所有密流都具备同等的存储分析价值。首先,密流内部价值分化显著。研究表明,绝大多数网络流量属于常规加密对话,仅不超过6.5%的密流包含潜在异常行为或新型特征[4]。若全量存储,有限的系统资源将被冗余数据占据,导致高价值密流在整体数据中的特征贡献被稀释。其次,密流价值随时间衰减。以传输层安全协议1.3版本(Transport Layer Security version 1.3, TLS 1.3)会话为例,其特征的有效分析窗口不超过24 h。随着时间推移,过期密流不仅占用有限的存储空间,还会削弱流量检测系统的动态适应能力。因此,一种可行的解决思路是在海量密流输入流量分析系统前,对其价值进行动态评估,并优先保留高价值样本,从而在有限资源约束下最大化密流检测效能。

在实际部署中,网关侧通常需对经过的流量进行轻量级解析与统计(如流级特征提取或聚合计数),以支持后续决策与分析。本文的目标是在此基础上进一步回答“哪些密流值得被长期保留并反复用于训练/更新”,通过对密流样本价值进行在线评估与优先级筛选,减少过期或低价值密流进入长期存储与建模流程,以提升资源利用效率与模型适应能力。

针对海量高速密流,流量分析系统普遍采用“先存后筛”的序贯策略[5-8],即先完整留存目标密流后,再从中选取代表性样本开展分析。此策略依赖人工设定的留存目标(如特定IP地址或端口),在网络环境持续演化的场景下,易导致已留存数据的边际价值逐步降低。即便人为提升留存目标多样性,由于缺乏系统化密流价值评估机制,不同留存数据仍难以实现价值沉淀,亦无法为后续的数据筛选与存储策略提供反馈优化。因此,面对动态演化的网络环境,流量分析系统应具备在海量密流传输过程中动态筛选的能力,即在密流持续流动变化中,实时评估密流样本的当前价值,以便在有限的存储资源下优先保留高价值密流,剔除陈旧或冗余密流。然而,这一目标的实现主要面临以下两方面挑战。

挑战一:密流模式高度异构,价值度量不易泛化。实际网络环境中,密流来源复杂多样,既包括IoT设

备产生的传感类密流，又包括来自手机等终端的交互类密流。不同类型的密流在数据包大小分布、时序特征及通信模式等方面差异显著。即使属于同一类型，不同实体的密流模式也会存在差异。跨类型、跨实体的密流模式高度异构性，使得构建统一的密流价值度量机制成为首要挑战。

挑战二：密流特征时间漂移，价值更新难以捕捉。随着IoT设备固件升级或App版本更新，密流特征逐渐发生漂移[9]。这种漂移不仅体现在数据包大小和时序等底层特征上，还可能波及上层密流语义，因此难以捕捉并与历史样本对齐。过期密流样本逐渐失去代表性，不仅占用存储空间，还会降低分析模型的动态适应能力。若密流价值度量机制缺乏漂移感知与动态适应能力，则其评估将逐渐偏离最优结果。

为解决以上挑战，本文提出了一种密流分析鲜度方法（FreshHunter）。其核心思想在于将密流样本的价值形式化为鲜度（Freshness），并从时间、空间与语义三个维度系统化建模。具体而言，在空间维度，FreshHunter从数据包、序列和流三个层级提取密流的分布与统计特征，刻画其结构特性。在语义维度，通过协议级与应用级交互建模，捕捉密流中包含的行为语义。二者结合可提炼出跨类型、跨行为的通用特征，实现异构密流的通用价值评估。在时间维度，FreshHunter则引入自适应更新机制，对密流鲜度予以时间衰减与周期性重估，以动态反映密流的时序演化及特征漂移，确保价值评估的时效性。依据密流鲜度建模结果，FreshHunter进一步构建了鲜度感知的对比学习框架。该框架引入可学习的聚类中心，引导密流样本嵌入表示在特征空间向类中心聚合，从而提升类内一致性与类间可分性，以确保密流样本群体多样性，并实现“边筛边存”的高鲜度密流样本在线留存机制。在密流留存过程中，FreshHunter将高鲜度密流的特征嵌入不断聚合至可学习类中心，以构建代表性密流特征的统一表征；同时结合特征漂移检测与自适应校正机制，动态更新类中心，以捕捉密流特征的时序性演化。由此，FreshHunter同时捕捉高价值、代表性的密流样本及其演化，促成了从瞬时数据到长期知识的沉淀，为下游各类密流分析任务提供前向知识沉淀、后向持续优化的训练数据支撑。

本文的主要贡献包括：

（1）提出了一种密流分析鲜度方法（FreshHunter），从时间、空间和语义三个维度形式化定义密流鲜度，动态评估异构密流的价值。

（2）构建了鲜度感知的对比学习框架，通过类中心的鲜度加权更新与损失函数优化，增强密流样本嵌入表示与类中心之间的一致性。

（3）实现了“边筛边存”的高鲜度密流样本在线留存，促成从瞬时数据捕获到长期知识累积的升级。

（4）在真实数据集上开展广泛实验评估，覆盖IoT终端、手机App与网站识别三类典型密流分析任务的九种主流分析算法，评估FreshHunter的有效性与通用性。

综上所述，本文针对在计算与存储资源受限场景下，流量分析系统难以实现海量密流在线建模与全量留存的问题，融合时间、空间和语义三维建模，提出了一种密流分析鲜度方法。该方法能够在海量异构密流的持续流动中实时评估样本价值，动态筛选并优先保留高鲜度、高价值的密流样本，促成从瞬时数据捕获到长期知识累积的升级，为提升密流分析的数据利用率与模型自适应能力等方面提供新的技术路径。

1 相关研究

随着加密通信的广泛应用[10]，流量分析研究已从传统的端口号和深度包检测[11]，逐步转向侧信道挖掘与机器学习建模[12-19]。然而，在计算与存储资源受限场景下，如何从高速海量密流中留存高价值样本，并据此训练高效的密流分析模型，仍是亟须解决的关键问题。现有研究主要围绕两个方向展开：一是高效密流分析，即基于不同粒度的密流特征（如数据包、序列、流）构建分析模型；二是密流价值评估与筛选，即从海量密流中筛选出高价值样本，以缓解数据冗余和模型训练开销。

1.1 密流分析

基于机器学习的密流分析方法，通过挖掘数据包大小、方向和时间等侧信道特征，无需解密数据包即可实现密流识别。具体而言，根据所采用的特征粒度，现有密流分析方法可分为三类。

（1）基于packet[12,20-22]。以单个数据包为最小分析单位，通过统计或机器学习方法捕捉单包分布规律。例如，Duan等人[12,20]利用双向数据包长度的频率分布对IoT设备进行分类。Dong等人[21]和Sun等人[22]则将数据包长度的统计特征作为IoT设备指纹。

（2）基于sequence[13-14,23-24]。将多个连续数据包组合成短期序列，以建模密流时序依赖关系，如序列包长分布、数据包间隔等。Trimananda等人[23]提出的PingPong能够自动从IoT设备密流中提取频繁出现的数据包长度和方向序列，作为设备指纹。Li等人[14]则从App密流中提取短期数据包突发，以捕捉特定App的结构性特征。

（3）基于flow[15-16,25-28]。以完整的传输控制协议/用户数据报协议（Transmission Control Protocol/User Da-

tagram Protocol,TCP/UDP)流为分析单位,从全局视角提取统计特征与时空特征。Ma 等人[16]通过构建不同流之间的上下文关联,实现轻量级代理下的网站识别。Liu 等人[26]则从双向流中提取时空特征,并输入卷积神经网络(Convolutional Neural Network,CNN)模型进行分类。

1.2　密流价值评估与筛选

原始密流通常包含大量冗余或噪声样本。这不仅增加了数据存储与模型训练的开销,还可能削弱密流分析模型的性能。因此,有必要在保证分析效果的前提下,筛选出具有代表性的高价值密流样本。现有密流筛选方法主要分为以下几类。

(1)基于特征重要性[5-6]。量化样本特征对分类结果的贡献,识别关键特征,并据此筛选高价值样本。常见方法包括决策树、随机森林和 Shapley 值。Zhang 等人[5]提出了一种加权对称不确定性 ROC(Receiver Operating Characteristic)曲线下面积算法(Weighted Symmetric Uncertainty-Area Under the ROC Curve,WSU_AUC)混合特征选择算法,用于筛选显著影响模型性能的特征。Muschalik 等人[6]将树模型沙普利加性解释(Tree SHAPley additive explanations,TreeSHAP)从单特征归因扩展到任意阶特征交互归因,提升了树模型中特征重要性的解释能力。

(2)基于流量相似度[7-8]。计算样本间相似性,减少冗余样本。常见的相似性度量方法包括动态时间规整(Dynamic Time Warping,DTW)、欧氏距离和分布差异等。Silva 等人[7]将多个短时间窗口内的数据包大小分布拼接成长序列,并利用 DTW 计算序列间相似度。

(3)基于信息熵[29]。衡量流量样本在统计分布中的信息量,评估其对整体流量多样性和代表性的贡献,并据此实现样本筛选。Williams 等人[29]提出一种基于香农熵聚合的流量分析方法,从数据包大小、包间隔时间、源/目的 IP 等关键特征提取熵,将高熵值流量判定为异常行为。

以上方法虽然在特征筛选与冗余控制方面取得一定成效,但普遍依赖静态数据分布或固定特征空间,难以适应网络环境动态演化和密流特征漂移。针对持续流动的海量异构密流,现有流量分析系统通常采用抽样存储与离线分析的策略,以固定或随机的方式抽取数据包/流,降低计算存储负担。

(1)随机采样[30-31]。对到达的数据包进行概率性抽取,以控制整体数据规模并维持统计意义上的代表性。典型实现包括等概率包采样、流采样以及基于时间间隔的周期性采样等,广泛应用于网络流(Network Flow,NetFlow)[30]、采样流(Sampled Flow,sFlow)[31]等流量监测系统中。

(2)基于规则的留存。通过预先设定的过滤条件或优先级规则(如特定 IP 地址、端口号、协议类型或通信方向等),选择性地保留满足条件的密流样本。

然而,上述抽样与留存机制主要侧重于控制样本规模以及保障统计意义上的总体代表性,缺乏对不同密流样本在长期分析价值上的精细化区分,因此难以有效判别哪些密流样本更值得被持续保留并反复用于模型训练或参数更新。一方面,随机或规则驱动的抽样策略在样本选择过程中未显式刻画密流中所蕴含的信息增量与行为新颖性,容易忽略在全局分布中占比低但分析价值较高的异常流量或新型密流模式;另一方面,此类机制通常依赖固定的抽样概率或静态规则运行,难以随网络环境变化及应用行为演化进行动态自适应调整,从而在非平稳流量场景下面临性能与数据利用效率的双重制约。

为此,本文提出一种密流分析鲜度方法,旨在动态演化的网络环境中实现对海量高速密流的价值实时评估与高价值样本筛选,并实现密流样本价值的长期沉淀。

2　密流分析鲜度方法

本章首先概述 FreshHunter 的整体架构,其次详细阐述密流鲜度动态评估、密流样本在线筛存以及自适应更新机制,最后从信息论视角分析方法运行机理。

2.1　整体架构

FreshHunter 的整体架构和工作流程如图 1 所示。算法输入为海量异构的原始密流,经鲜度价值动态评估,计算出密流样本的实时鲜度,并“边筛边存”各类别的高鲜度样本,以增强下游各类密流分析任务。

具体地,FreshHunter 从空间、时间和语义三个维度系统刻画密流价值,并据此计算其综合鲜度。在空间维度,算法基于不同粒度(数据包、序列、流)的密流统计与分布特征,提取密流结构差异;在语义维度,捕捉不同协议及应用的上下文行为模式,揭示其潜在语义;在时间维度,结合时间衰减与自适应更新机制,动态更新样本鲜度,以反映密流特征的时序变化。依据密流样本的鲜度计算结果,FreshHunter 对各类别密流样本进行实时排名,并构建鲜度感知的对比学习框架。经密流样本的嵌入表征与类内聚合、类间分离优化,优先保留类别集合中的高鲜度样本至容量为 N 的存储空间;同时,将留存样本的分布特征(如类别覆盖度与鲜度均值等)反馈至鲜度评估模块,用于校正空间与语义特征的建模基准,并自适应调整样本筛选阈值,以便在后续流量输入中持续优化鲜度计算与留存策略。最终,经过筛选与留存的密流不仅具备更高的信息价值,还保持跨类别的多样性,可直接作

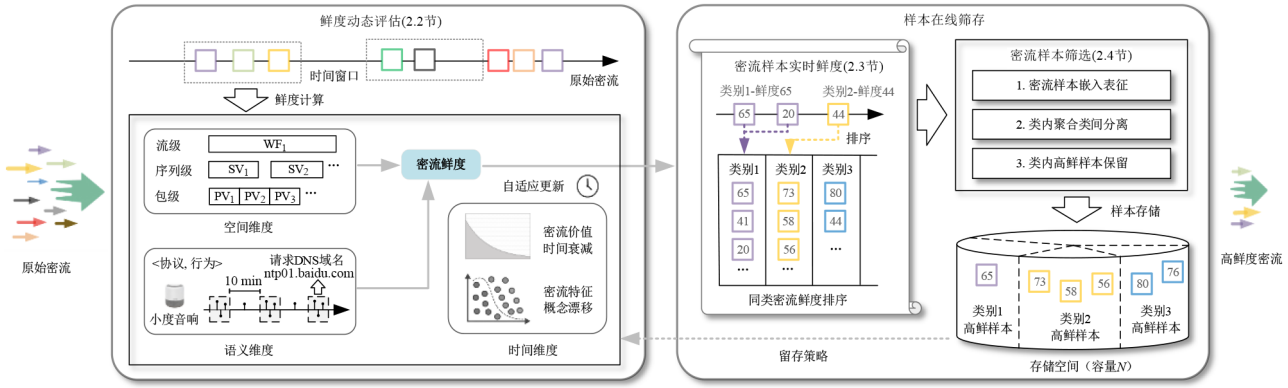


图1 FreshHunter的整体架构和工作流程

Figure 1 Overall architecture and workflow of FreshHunter

为IoT终端、手机App和网站识别等下游任务的高质量训练输入，显著提升密流分析模型的性能。

2.2 密流鲜度动态评估

网络流量在来源、协议及行为模式上高度异构，单一维度的特征刻画难以全面反映其价值。为此，本文从时间、空间与语义三个互补维度系统定义密流鲜度。其中，时间维度反映了密流样本的新旧程度及其时效性；空间维度刻画不同粒度（packet、sequence和flow）下密流特征的代表性；语义维度则捕捉不同协议或应用密流承载的上下文语义差异。

（1）时间维度。密流样本价值通常随时间推移而逐渐降低。为描述这一衰减规律，本文采用牛顿水冷定律[32]，定义时间维度的密流鲜度为

$$T_\varrho = \mathrm{e}^{-\lambda\left(t_c - t_f\right)} \tag{1}$$

其中，T_ϱ反映了密流样本在时间维度上的鲜度；λ为衰减系数；t_c为当前时间；t_f为密流样本ϱ的时间戳。式（1）表明，越接近当前时刻的密流，其鲜度值越高。

（2）空间维度。该维度旨在刻画不同粒度下，密流样本ϱ与整体分布之间的特征差异。具体而言，本文从三个特征粒度建模其空间鲜度。

packet。每个数据包被表征为向量**PV**，其形式为<数据包大小，数据包方向，时间戳>。

sequence。为了捕捉密流中频繁出现的短期burst，将多个连续的**PV**合并，构成序列向量**SV**。

flow。流向量**FV**基于流级统计特征构建，包括流持续时间、流长度、总数据包数、数据包平均大小等11维特征，以捕捉全局会话模式。

以上三类特征由同一密流样本ϱ派生，但处于不同的粒度层级与特征空间中，分别从包级局部结构、序列级中观结构到流级全局结构，对密流样本的空间特性进行互补刻画。基于此，本文将**PV**、**SV**、**FV**视为对同一密流样本的多粒度联合表征，则密流样本ϱ的空间特征可表示为

$$\chi_\varrho = \left\{\mathbf{PV}(\varrho),\ \mathbf{SV}(\varrho),\ \mathbf{FV}(\varrho)\right\} \tag{2}$$

为衡量ϱ与整体分布的差异，本文采用Kullback-Leibler散度（KL）定义其空间鲜度：

$$S_\varrho = D_{\mathrm{KL}}\left(P(\chi_\varrho),\ P_{\mathrm{ref}}^{\chi}\right) \tag{3}$$

其中，D_{KL}为两者之间的分布差异；$P(\chi_\varrho)$为样本ϱ的空间特征分布；P_{ref}^{χ}为整体特征分布。式（3）表明，样本与整体密流分布的空间特征差异越大，其空间鲜度值越高，意味着在训练过程中该样本更可能提供新的信息；反之，其空间维度价值有限。

在实际网络中，受IoT设备固件升级、手机App版本更新等因素影响，密流特征往往发生显著漂移[33-34]。由于空间鲜度基于样本分布和整体分布的差异度量，其计算过程具有天然的抗漂移性。具体而言，整体密流特征分布P_{ref}^{χ}会随时间动态更新，使空间鲜度能够反映样本与最新密流模式之间的差异。当历史样本逐渐偏移当前分布，其空间鲜度降低并被筛除；而新产生的、在当前分布下具有代表性的样本则获得更高鲜度值，并被优先保留。

（3）语义维度。不同协议在网络通信中承载着差异化的上下文语义，因此其分析价值也不相同。例如，小度智能音箱每隔10 min访问一次域名ntp01.baidu.com，以校准时间。又如，动态主机配置协议（Dynamic Host Configuration Protocol，DHCP）记录了设备接入网络时的配置信息。基于此，本文将密流语义特征建模为

$$\xi_\varrho = \left\{\mathrm{Protocol},\ \mathrm{Context}\right\} \tag{4}$$

其中：Protocol代表协议类别，如DHCP、域名系统（Domain Name System，DNS）、简单服务发现协议（Simple Service Discovery Protocol，SSDP）等；Context代表协议所处的具体上下文。例如，在DNS协议下，Context为具体查询域名；在DHCP协议下，Context为Options字段值。

为度量语义层面的差异性，密流语义鲜度定义为

$$M_\varrho = D_{\mathrm{KL}}\left(P(\xi_\varrho),\ P_{\mathrm{ref}}^{\xi}\right) \tag{5}$$

其中,$P(\xi_\varrho)$为样本ϱ的语义特征分布,P_{ref}^{ξ}为整体密流的语义特征分布。

综合考虑时间、空间和语义三个维度,本文定义密流样本ϱ的鲜度为

$$F_\varrho=\exp\left(\alpha\ln T_\varrho+\beta\ln(\widetilde{S_\varrho}+\varepsilon)+\gamma\ln(\widetilde{M_\varrho}+\varepsilon)\right) \tag{6}$$

其中,$\widetilde{S_\varrho}$和$\widetilde{M_\varrho}$分别为空间和语义维度经尺度压缩后的鲜度,$\alpha+\beta+\gamma=1$,$\varepsilon\in(0,10^{-3}]$。

2.3　鲜度感知的对比学习

鲜度度量保证了单个密流样本的代表性,整体密流样本还需保持多样性。为此,本文提出一种鲜度感知的对比学习方法,用于筛选兼具高鲜度与多样性的密流样本。

如图2所示,该方法通过引入可学习的聚类中心,实现类内样本聚合,类间样本分离。以IoT设备密流为例,算法首先在密流鲜度提取阶段对输入样本进行多维度建模,从时间、空间与语义三维特征中计算每个样本的鲜度得分。其次,在嵌入表征增强阶段,计算样本间的特征距离,构建鲜度加权的类中心。最后,在类别分离阶段,通过鲜度感知的对比学习模型优化嵌入空间表征,使样本在距离与鲜度分布上同时达到最优,从而实现对高价值密流样本的优先筛选。

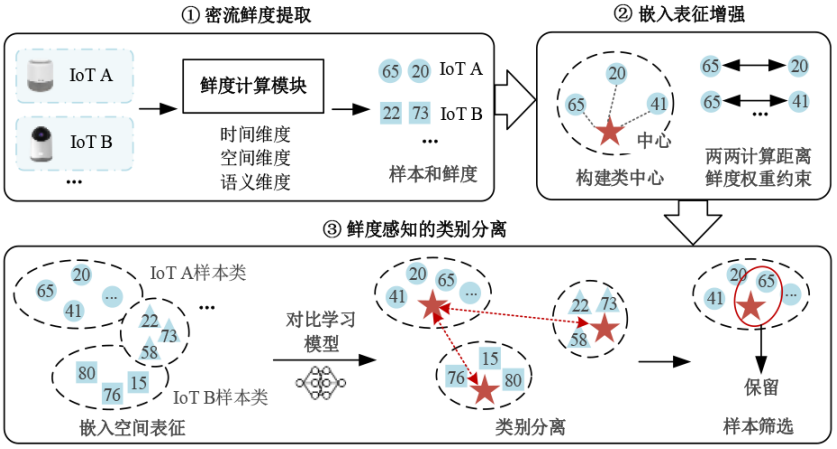


图2　基于鲜度感知对比学习的密流筛选

Figure 2　Freshness-aware contrastive learning for encrypted traffic filtering

(1)正负样本对构建。在训练对比学习模型时,首先需构建正负样本对。正样本对由同一类别下的不同样本构成,负样本对则来自不同类别,具体如下:

$$\begin{cases}(\varrho_a,\varrho_p):\ \text{Label}(\varrho_a)=\text{Label}(\varrho_p)\\(\varrho_a,\varrho_n):\ \text{Label}(\varrho_a)\neq\text{Label}(\varrho_n)\end{cases} \tag{7}$$

其中,ϱ_a为锚点样本,ϱ_p为正样本,ϱ_n为负样本。

(2)鲜度感知的类中心。为增强类中心的自适应性,本文将密流鲜度F_ϱ融入类中心计算。设类别y下的样本集合为C_y,其类中心$\boldsymbol{c}_y$可表示为

$$\boldsymbol{c}_y=\frac{\sum_{\varrho_i\in C_y}F_{\varrho_i}\boldsymbol{h}(\varrho_i)}{\sum_{\varrho_i\in C_y}F_{\varrho_i}} \tag{8}$$

其中,$\boldsymbol{h}(\varrho_i)$为样本嵌入表示。为获得统一且结构化的特征表达,本文采用基于深度神经网络的嵌入架构,将原始密流样本ϱ_i映射至统一的低维嵌入空间$\boldsymbol{h}(\varrho_i)$。具体地,首先提取样本$\varrho_i$的空间和语义特征,然后对各维特征进行拼接与归一化处理,并作为输入送入嵌入网络,以学习紧凑且具有判别性的向量表示。嵌入网络采用轻量化的多层感知机结构,由多层全连接层、非线性激活函数及批量归一化模块组成。网络末层输出固定维度的嵌入向量$\boldsymbol{h}(\varrho_i)\in\mathbb{R}^d$,其中嵌入维度$d$设为128。

鲜度值越高,样本在类中心计算中的权重越大,从而保证了类中心能够随密流分布的动态变化自适应调整。

(3)损失函数定义。传统对比学习依赖三元组$\{\varrho_a,\varrho_p,\varrho_n\}$优化样本间的相对距离。然而,在实际网络中,受卡顿与网络延迟等因素影响,同类样本间可能存在较大偏差,从而导致不同类别的特征分布重叠,削弱了三元组方法的聚类效果。

为此,本文采用类中心约束的对比学习损失,优化样本与类中心的距离。设类中心集合为$\{\boldsymbol{c}_1,\boldsymbol{c}_2,\cdots,\boldsymbol{c}_y\}$,样本$\varrho_i$的嵌入表示为$\boldsymbol{h}(\varrho_i)$,其类别标签为$l\in\{1,2,\cdots,y\}$。样本$\varrho_i$的损失函数定义为

$$L(\varrho_i,l)=-F(\varrho_i)\cdot\ln\frac{\mathrm{e}^{-\alpha D(\boldsymbol{h}(\varrho_i),\boldsymbol{c}_l)}}{\mathrm{e}^{-\alpha D(\boldsymbol{h}(\varrho_i),\boldsymbol{c}_l)}+\sum_{j\neq l}\mathrm{e}^{-\alpha\left(D(\boldsymbol{h}(\varrho_i),\boldsymbol{c}_j)-\delta\right)}} \tag{9}$$

其中,$D(\cdot,\cdot)$为距离度量函数;δ为类间margin,用于增强类别分离性。本文将δ作为全局超参数,在验证集上进行调优,以在类内紧致性与类间分离性之间取得

平衡。α为温度缩放系数，用于调节距离差异在损失函数中的影响程度。在$\alpha\in[5,20]$、$\delta\in[0.3,0.7]$时，模型性能保持稳定。

为防止不同类中心过度聚合，引入了正则化项：

$$\Omega(\boldsymbol{c})=\beta\sum_{j\neq k}\frac{1}{\left\|\boldsymbol{c}_j-\boldsymbol{c}_k\right\|^2} \tag{10}$$

最终，对比学习模型的目标为最小化全体样本的损失之和。

$$\min_{\{\boldsymbol{c}_j\}} E_{(\varrho,l)\sim P}\left[L\left(\varrho,l\right)\right]+\Omega(\boldsymbol{c}) \tag{11}$$

2.4 密流采样与留存

完成密流鲜度建模与嵌入空间优化后，仍需在有限存储空间内实现样本的动态留存。为此，本文提出一种双阶段留存方法：第一阶段为类别采样，根据各类别密流的到达速率与特征不确定性，动态分配采样配额N_c；第二阶段为样本留存，在类别内部依据样本鲜度优先级动态维护存储集合R_c，以在有限空间内优先保留高鲜度样本。通过类别级采样与样本级留存的协同决策，实现高鲜度密流样本的动态留存。

（1）类别采样。在时间窗口$[t-\Delta,t]$内，设类别$\boldsymbol{c}$的密流样本到达数量为n_c，到达速率为ρ_c，样本鲜度方差为$\mathrm{Var}_c(F)$。为实现自适应配额分配，本文综合考虑样本速率与不确定性，定义类别$\boldsymbol{c}$的采样配额为

$$N_c\propto\left(\rho_c+\lambda_1\tilde{\rho}_c+\lambda_2\widetilde{\mathrm{Var}}_c(F)+\lambda_3\tilde{\zeta}_c\right) \tag{12}$$

其中：$\tilde{\rho}_c$和$\widetilde{\mathrm{Var}}_c(F)$分别为归一化后的到达速率与鲜度方差；$\tilde{\zeta}_c$为可调节的类别优先系数，用于在特定场景下提升特定类别（如冷门类别密流）的权重。

类别级配额采用周期性重计算、窗口内冻结不变的动态更新策略。具体而言，系统在滑动时间窗口$[t-\Delta,t]$内统计各类别的到达速率与鲜度分布特征，并基于该窗口内的统计结果计算类别配额N_c。N_c仅在窗口结束时统一触发一次更新。更新周期Δ的设定与实际流量负载水平密切相关。在本文实验设置中，Δ取分钟级时间尺度。一方面，该机制通过周期性重分配机制抑制热门类别在高峰期对缓存空间的过度占用；另一方面，为冷门类别保留具有代表性的样本，避免其在竞争中被持续挤压。

（2）样本留存。对于每个类别$\boldsymbol{c}$，系统维护容量为N_c的留存集合R_c，用于存储最具代表性和分析价值的密流样本。对每个到达样本ϱ_i，根据式（6）计算其综合鲜度F_i。为在保持采样公平性的同时提升高鲜度样本被选中的采样概率，定义其优先级为

$$p_i=\frac{F_i}{u_i},\ u_i\sim\mathrm{U}(0,1] \tag{13}$$

其中，u_i为从区间$(0,1]$均匀采样的随机变量。系统维护按p_i升序排序的最小堆结构，以实现流式高效更新。留存策略如下：当$|R_c|<N_c$时，样本ϱ_i直接加入集合；否则，若$p_i>p_{\min}(R_c)$，则进入候选替换阶段，样本ϱ_i替换堆顶样本；若$p_i<p_{\min}(R_c)$，则丢弃样本。

该策略可确保样本留存的概率与其鲜度F_i成正比，使系统在有限空间内优先保留高鲜度、高价值样本，同时持续淘汰陈旧或冗余样本，实现密流样本的持续优化留存。由于每个类别对应的堆容量始终受限于N_c，样本级更新仅涉及一次优先级计算以及一次$O(\log N_c)$的堆操作，其时间、空间开销均与总体流量规模无关，因此能够在高速密流输入条件下保持良好的吞吐性能。

2.5 运行机理分析

FreshHunter旨在有限存储空间下，最大化地保留密流样本对分析模型的信息贡献。基于式（6）定义的密流样本鲜度F_ϱ，首先将其归一化，得到全体密流样本的鲜度分布。

$$P(F_\varrho)=\frac{F_\varrho}{\sum\limits_{\varrho\in\mathcal{D}}F_\varrho} \tag{14}$$

在此基础上，可定义全体密流集合$\mathcal{D}$的鲜度熵为

$$H_{\mathrm{all}}=-\sum_{\varrho\in\mathcal{D}}P(F_\varrho)\ln P(F_\varrho) \tag{15}$$

当FreshHunter筛选出子集$\mathcal{D}'\subseteq\mathcal{D}$后，$\mathcal{D}'$的鲜度熵可表述为

$$H_{\mathrm{FF}}=-\sum_{\varrho\in\mathcal{D}'}P'(F_\varrho)\ln P'(F_\varrho) \tag{16}$$

其中，$P'(F_\varrho)=\dfrac{F_\varrho}{\sum\limits_{\varrho\in\mathcal{D}'}F_\varrho}$。$H_{\mathrm{FF}}$与$H_{\mathrm{all}}$越接近，说明FreshHunter在样本大幅压缩的情况下，依然能够保持原始密流分布的多样性与不确定性。换言之，被剔除的样本大多属于低熵贡献的冗余数据，其对密流分析任务的边际信息增益趋近于零，而保留下来的样本则在时间、空间和语义维度上具有更高的信息价值。

为验证上述推断，本文基于公开的IoT密流数据集UNSW（UNSW-IoT dataset）[35]和文献[36]中的密流分析算法开展实验，对Random、DTW、TreeSHAP及FreshHunter四种密流筛选算法的性能进行了系统评估。具体而言，四种筛选算法分别保留10%~100%的原始密流样本，并计算相应的H_{FF}用于比较。本文采用三类信息熵指标进行定量分析。

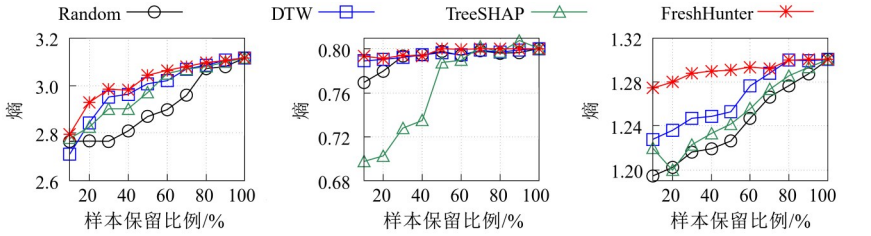
（1）标签类别熵，用于衡量筛选子集在类别分布上的均匀性，熵值越高，表明所保留样本的类别越全面、结构越均衡。

（2）窗口平均熵，用于评估单个时间窗口内密流样本的特征复杂度与多样性，反映筛选后样本的丰富

程度。

(3)全局分布熵,用于衡量筛选子集在整体样本空间中的覆盖度,熵值越高,表示子集保留了更多原始密流的信息结构。

实验结果如图3所示。整体而言,随着筛选比例的逐步提升,Random、DTW、TreeSHAP和FreshHunter所保留样本的信息熵均呈现上升趋势。其中,FreshHunter的三种信息熵均优于其余三种筛选算法。如图3(a)所示,FreshHunter在所有采样比例下均保持最高的标签类别熵,尤其在低比例区间(10%~40%)中的提升最为显著,这说明该方法能够在有限的样本预算下有效维持类别分布的完整性与均衡性。如图3(b)所示,FreshHunter和DTW的窗口平均熵表现较为稳定,显著优于TreeSHAP的窗口平均熵,这表明其在局部时间窗口内能够保留更高的特征复杂度。如图3(c)所示,FreshHunter在所有采样比例下的全局分布熵始终保持在最高水平,显著超过Random、DTW和TreeSHAP的全局分布熵,这表明其在样本压缩时仍能最大程度保留原始密流的全局分布结构。



(a) 标签类别熵　(b) 窗口平均熵　(c) 全局分布熵

(a) Label class entropy　(b) Window average entropy　(c) Global distribution entropy

图3　不同筛选比例下四种密流筛选算法保留样本的信息熵

Figure 3　Information entropy of retained samples across filtering ratios for four encrypted traffic filtering algorithms

3　实验

本文实验旨在回答以下三个研究问题。

(1)研究问题一:FreshHunter能够在多大程度上提升密流分析模型的性能?

(2)研究问题二:面对密流特征漂移,FreshHunter能否保持有效性?

(3)研究问题三:时间、空间与语义三个维度分别如何影响FreshHunter的整体性能?

3.1　数据集

为回答以上研究问题,本文实验涵盖了三类典型密流分析任务:IoT设备、手机App以及网站(Website)识别,并分别在对应数据集上开展评估。

UNSW(IoT)。该数据集由新南威尔士大学的Sivanathan等人[35]历时200余天收集,涵盖了28种不同类型的IoT设备,包含摄像头、台灯、传感器等。

Android150(App)。该数据集由Sungkyunkwan大学的Oh等人[27]收集,涵盖来自Google Play商店最受欢迎的150个App,如Facebook、Instagram、Airbnb等。

文献[37](Website)。该数据集由Waterloo大学的Wang等人[37]通过Tor浏览器采集,包含监控集(100个敏感网站,每个网站采集90次访问实例)和非监控集(Alexa Top 10000的9 000个常见网站,每个网站采集1次访问实例)。

文献[38](Website概念漂移)。该数据集由Ku Leuven大学的Rimmer等人[38]历时2个月收集,涵盖Alexa排名前200的热门网站。初始阶段为每个网站收集2 000条访问样本,并分别在3 d、10 d、2周、4周和6周后再次采集100条样本,用于评估网站内容动态变化对指纹识别精度的影响。

3.2　基线算法

FreshHunter可无缝兼容各种基于机器学习的密流分析算法。为实现全面的性能评估,本文将FreshHunter集成至三类主流密流分析任务的九种典型算法中。具体算法见表1。

为公平评估FreshHunter的增强效果,本文将三种典型的密流筛选方法集成到上述分析算法中,作为对比基线。

(1)Random。从密流样本中均匀随机抽取一定比例样本。

(2)DTW[7]。利用DTW度量密流序列间相似度。若新样本与已保留样本的DTW距离低于预设阈值,则视为冗余并过滤;反之则保留。

(3)TreeSHAP[6]。基于密流特征训练TreeSHAP模型,对特征重要性进行量化,并根据SHAP值对样本重要性排序,优先保留特征贡献度较高的样本。

3.3　实验环境与设置

本文所有实验均在同一台Inspur NF5468M5服务器上完成,操作系统为Ubuntu 22.04.5 LTS。硬件环境

表1 密流分析算法与原理

Table 1 Encrypted traffic analysis algorithms and their principles

分析任务	分析算法	分析原理
IoT	文献[36]	基于双向流的时空特征,特别是初始n个数据包的长度与时间间隔,识别IoT设备
	文献[39]	采用统计特征捕捉短时间内的数据包动态,包括数据包总长度、平均长度以及长度偏差
	文献[26]	基于卷积神经网络,对数据包长度序列进行建模,实现IoT设备指纹识别
App	AppScanner[40]	基于密流突发的统计特征,包括数据包长度的最大值、最小值、平均值、标准差、方差和偏斜度等
	AppSniffer[27]	采用一种双阶段方法识别App密流。第一阶段基于TCP流的数量区分正常密流与VPN密流;第二阶段利用LightGBM与FastAI的堆叠集成模型学习数据包的时空特征
	MCS[41]	从TCP流突发中提取统计特征,如数据包长度、到达间隔时间、上下行比例与方向变化等,并结合贝叶斯、随机森林等多种分类器完成App识别
Website	k-FP[42]	基于随机森林叶子节点构建密流指纹向量,并利用汉明距离计算指纹相似度,最终结合kNN分类实现网站识别
	DF[43]	将数据包方向序列输入卷积神经网络,通过多层卷积、批量归一化与RELU激活函数自动提取特征,实现网站指纹识别
	RF[44]	通过构建密流聚合矩阵(在时间片内统计进出数据包数量),提取对填充与延迟不敏感的稳健特征,并利用卷积神经网络学习高层密流模式,实现网站识别

如下:Intel Xeon Gold 6248R处理器(主频3.00 GHz)、NVIDIA GeForce RTX 3080 GPU(10 GB显存)以及1 TB DDR4 ECC RDIMM内存。代码运行环境为Python 3.10.4,深度学习框架采用PyTorch 2.7.1。FreshHunter中鲜度感知对比学习模型的超参数设置如下:采用Adam优化器,初始学习率为1×10^{-3},Batch Size为256,训练轮数为100,嵌入维度为128,weight decay为5×10^{-4}。

FreshHunter的在线处理模块仅依赖流级特征与简单的序列级统计信息,并结合轻量化嵌入计算完成分析,其整体处理流程在系统架构层面与NetFlow、sFlow等主流流量监测系统高度一致,即均在流量经过网关或采集节点时完成一次快速的特征提取与筛选决策。FreshHunter不涉及深度包检测或复杂的内容解析过程,其在线处理复杂度随流量规模呈线性增长,主要受特征维度规模与嵌入网络规模的影响。从系统设计看,FreshHunter可作为独立的分析模块与现有流量采集或采样组件并行部署,并直接复用其输出结果作为输入,在不显著降低整体处理吞吐率的前提下,实现对高价值密流样本的精细筛选与长期留存。

3.4 实验一:FreshHunter性能评估

(1)实验设置。本实验评估了Random、DTW、TreeSHAP和FreshHunter四种密流筛选方法对九种密流分析算法的性能提升效果。分析算法涵盖IoT、App和Website这三类主流的密流分析任务,分别对应UNSW[35]、Android150[27]以及文献[37]数据集。为避免类别配额差异带来的偏差,实验对各类别密流采用统一比例的样本留存方式。四种密流筛选方法从各数据集的训练集中按比例(10%~100%)抽取样本用于模型训练,并在测试集上评估,最终以检测模型的F_1分数作为性能指标。

(2)实验结果。图4展示了Random、DTW、TreeSHAP和FreshHunter四种密流筛选算法在IoT、App和Website三类密流分析任务、九种分析算法上的性能表现。总体来看,随着样本筛选比例的提升,各分析算法的F_1分数均呈现递增趋势。在相同样本比例下,FreshHunter的表现始终优于其他三种方法,能够更有效地保留高价值密流样本并提升分析模型性能。

在IoT检测任务中,FreshHunter相较于Random、DTW和TreeSHAP,平均提升约15%~40%的分析性能。以文献[36]的分析算法为例,当样本保留比例为10%时,FreshHunter的F_1分数为0.31,而Random、DTW和TreeSHAP分别仅为0.15、0.27和0.21。在App分析任务中,FreshHunter同样展现出显著优势。当样本保留比例为10%时,在AppScanner算法上,其F_1分数较Random、DTW和TreeSHAP分别提高111%、16%和50%;在AppSniffer算法上,提升幅度分别为142%、61%和134%;在MCS算法上,分别提升103%、19%和13%。在Website识别任务中,FreshHunter同样优于其他三种筛选算法,特别是在低样本比例区间(10%~40%)。在k指纹识别(k-FingerPrinting,k-FP)算法上,FreshHunter分别较Random、DTW和TreeSHAP提升7%、4%和20%;在深度指纹识别(Deep Fingerprinting,DF)算法上,分别提升40%、8%和21%;在鲁棒指纹识别(Robust Fingerprinting,RF)算法上,分别提升35%、18%和24%。

(3)研究问题一的结论。在三类密流分析任务及九种分析算法的评估中,FreshHunter均能有效筛选高价值样本并显著提升密流分析算法的性能。与Random、DTW和TreeSHAP相比,FreshHunter在各种筛选

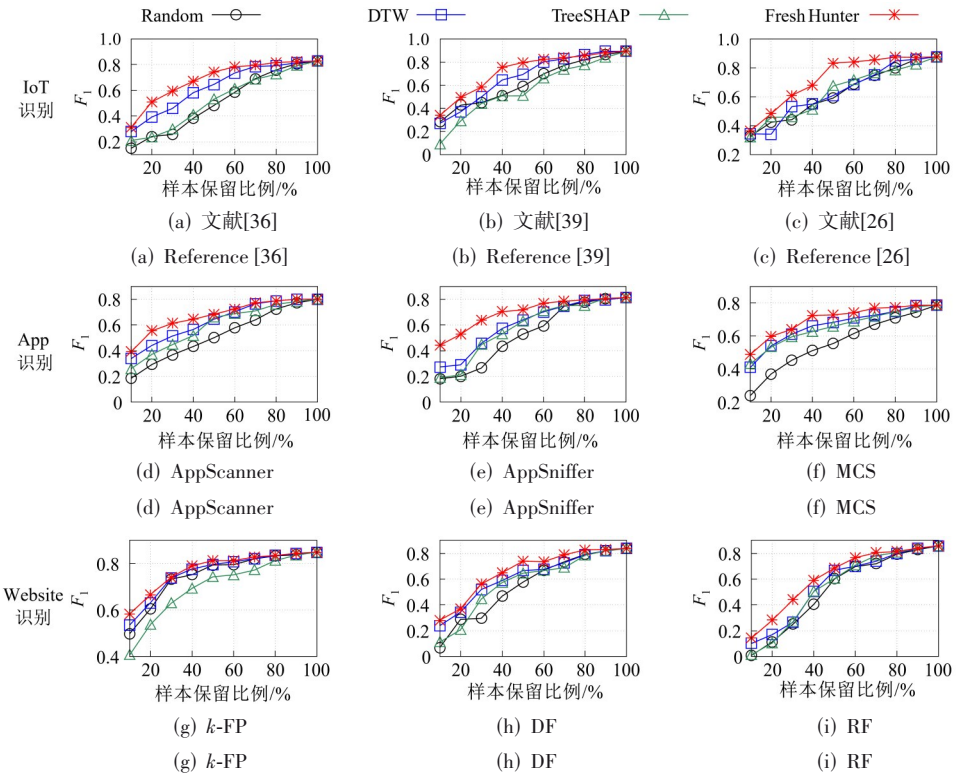


图4　四种密流筛选方法在三类密流分析任务中的性能对比

Figure 4　Performance comparison of four encrypted traffic filtering methods across three encrypted traffic analysis tasks

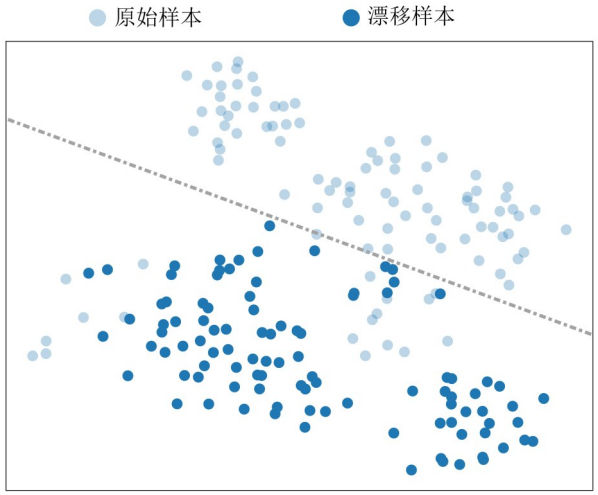
比例下始终保持最优表现,平均提升分析算法的 F_1 分数约15%~60%。

3.5　实验二:FreshHunter抗漂移评估

(1)实验设置。本实验评估了Random、DTW、TreeSHAP和FreshHunter四种密流筛选方法在特征漂移场景下的性能。实验选取Website识别场景分析。与IoT和App相比,Website密流的模式漂移现象更为显著。所使用的检测算法为DF[43],数据集为文献[38]数据集,其中两周前采集的Website密流被视为原始密流,两周后采集的密流被视为漂移密流。原始密流与漂移密流中各取1/3作为训练集,其余2/3漂移密流作为测试集。在模型训练阶段,四种密流筛选方法分别从训练集中按比例(10%~100%)筛选样本进行模型训练,并在测试集上进行性能评估。

(2)实验结果。如图5所示,基于文献[38]数据集,对原始密流样本和漂移密流样本进行 t-SNE 特征空间可视化映射。其中,深蓝色点表示漂移后样本,浅蓝色点表示原始样本。由图5可知,两类样本在特征空间中呈现显著分离,漂移样本的整体分布发生明显偏移,形成新的聚类区域。

如表2所示,在密流特征漂移场景下,FreshHunter展现出显著优于其他密流筛选算法的自适应能力。当使用100%的训练数据时,分析算法DF的 F_1 分数为0.74。将Random、DTW和TreeSHAP三种筛选算法分别集成至DF算法后,DF仍难以区分漂移前后的密流特征,只能将两者样本一并用于训练,导致引入大量无效特征,限制了模型的泛化能力并导致分析性能下降。尤其在低筛选比例(10%~30%)下,Random的 F_1 分数仅为0.05~0.20,DTW为0.08~0.38,而



注:浅色点表示历史密流样本,深色点表示后续时间窗口出现的漂移样本。

图5　Website密流特征漂移

Figure 5　Feature drift in website encrypted traffic

TreeSHAP 仅为 0.01~0.08。相比之下，FreshHunter 通过引入自适应更新机制，能够动态感知密流分布的时序漂移并优先保留高鲜度样本，从而显著提升分析模型在漂移场景下的学习效率与稳定性。在筛选比例为 50% 时，FreshHunter 能够准确识别漂移样本，将 DF 分析算法的 F_1 分数提高到 0.89。

（3）研究问题二的结论。FreshHunter 能够自适应密流特征漂移，将 DF 分析算法的 F_1 分数提升约 20%。

表 2 不同密流筛选算法的抗漂移性能对比

Table 2 Comparison of drift robustness across different encrypted traffic filtering algorithms

筛选比例	10%	20%	30%	40%	50%	60%	70%	80%	90%	100%
Random	0.05	0.21	0.20	0.37	0.52	0.56	0.57	0.67	0.72	0.74
DTW	0.08	0.28	0.38	0.47	0.64	0.65	0.69	0.73	0.73	0.74
TreeSHAP	0.01	0.04	0.08	0.17	0.28	0.39	0.53	0.68	0.72	0.74
FreshHunter	**0.45**	**0.74**	**0.80**	**0.84**	**0.89**	—	—	—	—	—

注：加粗字体为最优结果。

3.6 实验三：消融实验

（1）实验设置。本实验评估了 FreshHunter 中时间、空间与语义三个维度对其性能的影响。实验基于 UNSW[35] 数据集开展，选取文献[36]、文献[39]和文献[26]三种典型 IoT 分析算法作为下游模型。为分析各维度的独立贡献，构建了 FreshHunter 的四种变体：完整模型、去除空间维度、去除语义维度以及去除时间维度。四种变体均从训练集中按 40% 的比例筛选样本（此比例下模型性能已趋于稳定，且差异具有区分度），并据此进行模型训练与性能评估。

（2）实验结果。如表 3 所示，包含完整空间、语义与时间三维鲜度建模的 FreshHunter，在集成至文献[36]、文献[39]和文献[26]三种分析算法后，分别取得了 0.67、0.75 和 0.68 的 F_1 分数。从各维度的独立影响来看，空间维度对算法性能的贡献最为显著。当空间维度被移除后，三种分析算法的 F_1 分数分别下降至 0.49、0.43 和 0.39，相较完整算法下降幅度达 27%、43% 和 43%。这一结果表明，空间特征在表征密流的结构差异与分布多样性方面发挥了关键作用，是区分密流类别的核心因素。语义维度对 FreshHunter 性能亦有显著影响。当语义维度被移除时，F_1 分数分别降至 0.54、0.59 和 0.52，下降幅度为 19%~24%。这说明协议级与应用级的行为语义特征能够有效补充空间特征，通过捕捉上下文关联与交互模式，增强了样本筛选的判别能力与密流分类的语义一致性。相较之下，时间维度在本实验场景中的作用相对有限。当时间维度被移除时，F_1 分数分别为 0.64、0.74 和 0.55，整体性能变化较小。这主要是由于实验数据集的采样周期较短，密流分布在训练与测试阶段保持相对稳定，未能充分体现密流时间漂移特征。在实验二的漂移场景中，时间维度对密流特征漂移的自适应调节效果显著，在动态演化场景下具有重要价值。

表 3 FreshHunter 消融实验

Table 3 Ablation study of FreshHunter

空间	语义	时间	文献[36]	文献[39]	文献[26]
×	√	√	0.49	0.43	0.39
√	×	√	0.54	0.59	0.52
√	√	×	0.64	0.74	0.55
√	√	√	0.67	0.75	0.68

（3）研究问题三的结论。在 FreshHunter 中，空间维度对其性能的提升最为显著，其次为语义维度，时间维度则在动态场景中具备自适应优势。

4 结论

本文针对在存储与计算资源受限条件下，流量分析系统难以实现海量密流全量留存与分析的问题，提出了一种密流分析鲜度方法——FreshHunter。该方法融合时间、空间和语义三个维度，将密流样本的价值形式化为鲜度，并据此构建鲜度感知的对比学习框架，筛选出兼具高鲜度与多样性的密流样本，进而促成从瞬时数据捕获到长期知识累积的升级。本文在真实数据集上开展广泛评估，验证 FreshHunter 在 IoT 设备、手机 App 和网站识别三类典型密流分析任务及九种分析算法上的性能。实验结果表明，该方法可提升九种分析算法的 F_1 分数约 15%~60%。在密流特征漂移场景下，FreshHunter 能够提升分析算法的 F_1 分数约 20%。该研究为资源受限条件下增强密流分析的模型性能、自适应能力与数据利用率等方面提供新的技术路径。未来，将进一步探索 FreshHunter 的应用潜力，包括更广泛的加密流量分析场景（如恶意流量检测、入侵检测与异常行为识别）、更鲁棒的自适应概念漂移场景（如跨网络环境和跨设备类型等）、更大规模下的工程优化（如支持更高频率的在线更新，实现关键参数的自动化调节），以提升其在真实网络环境中的长期稳定运行能力。

参考文献

[1] Antonakakis M, April T, Bailey M, et al. Understanding the mirai botnet[C]//Proceedings of the 26th USENIX Conference on Security Symposium. Berkeley: USENIX Association, 2017: 1093-1110.

[2] 沙泓州, 刘庆云, 柳厅文, 等. 恶意网页识别研究综述[J]. 计算机学报, 2016, 39(3): 529-542.
Sha Hongzhou, Liu Qingyun, Liu Tingwen, et al. Survey on malicious webpage detection research[J]. Chinese Journal of Computers, 2016, 39(3): 529-542. (in Chinese)

[3] 李艳, 王纯子, 黄光球, 等. 网络安全态势感知分析框架与实现方法比较[J]. 电子学报, 2019, 47(4): 927-945.
Li Yan, Wang Chunzi, Huang Guangqiu, et al. A survey of architecture and implementation method on cyber security situation awareness analysis[J]. Acta Electronica Sinica, 2019, 47(4): 927-945. (in Chinese)

[4] Belson D. Cloudflare 2024 年度回顾[EB/OL]. (2024-12-09)[2026-02-23]. https://blog.cloudflare.com/zh-cn/radar-2024-year-in-review/.
Belson D. Cloudflare 2024 year in review[EB/OL]. (2024-12-09)[2026-02-23]. https://blog.cloudflare.com/zh-cn/radar-2024-year-in-review/. (in Chinese)

[5] Zhang Hongli, Lu Gang, Qassrawi M T, et al. Feature selection for optimizing traffic classification[J]. Computer Communications, 2012, 35(12): 1457-1471.

[6] Muschalik M, Fumagalli F, Hammer B, et al. Beyond Tree-SHAP: Efficient computation of any-order Shapley interactions for tree ensembles[J]. Proceedings of the AAAI Conference on Artificial Intelligence. Washington: AAAI Press, 2024, 38(13): 14388-14396.

[7] Silva P R, Vinagre J, Gama J. A DTW approach for complex data a case study with network data streams[C]//Proceedings of the 38th ACM/SIGAPP Symposium on Applied Computing. New York: ACM, 2023: 402-409.

[8] Chung J Y, Park B, Won Y J, et al. An effective similarity metric for application traffic classification[C]//Proceedings of 2010 IEEE Network Operations and Management Symposium. Piscataway: IEEE, 2010: 286-292.

[9] 程光, 钱德鑫, 郭建伟, 等. 基于散度的网络流概念漂移分类方法[J]. 计算机研究与发展, 2020, 57(12): 2673-2682.
Cheng Guang, Qian Dexin, Guo Jianwei, et al. A classification approach based on divergence for network traffic in presence of concept drift[J]. Journal of Computer Research and Development, 2020, 57(12): 2673-2682. (in Chinese)

[10] Google. Google transparency report: HTTPS overview[EB/OL]. [2026-02-23]. https://transparencyreport.google.com/https/overview.

[11] Cao Zigang, Cao Shoufeng, Xiong Gang, et al. Progress in study of encrypted traffic classification[C]//Proceedings of the International Conference on Trustworthy Computing and Services. Berlin, Heidelberg: Springer, 2013: 78-86.

[12] Duan Chenxin, Gao Hao, Song Guanglei, et al. ByteIoT: A practical IoT device identification system based on packet length distribution[J]. IEEE Transactions on Network and Service Management, 2021, 19(2): 1717-1728.

[13] Li Jianfeng, Wu Shuohan, Zhou Hao, et al. Packet-level open-world app fingerprinting on wireless traffic[C]//Proceedings of the 29th Annual Network and Distributed System Security Symposium. Reston: Internet Society, 2022: 24210.

[14] Li Jianfeng, Zhou Hao, Wu Shuohan, et al. FOAP: Fine-grained open-world android app fingerprinting[C]//Proceedings of the 31st USENIX Security Symposium. Berkeley: USENIX Association, 2022: 1579-1596.

[15] Ma Xiaobo, Qu Jian, Li Jianfeng, et al. Pinpointing hidden IoT devices via spatial-temporal traffic fingerprinting[C]//Proceedings of the IEEE Conference on Computer Communications. Piscataway: IEEE, 2020: 894-903.

[16] Ma Xiaobo, Shi Mawei, An Bingyu, et al. Context-aware website fingerprinting over encrypted proxies[C]//Proceedings of the IEEE Conference on Computer Communications. Piscataway: IEEE, 2021: 9488676.

[17] 谷勇浩, 徐昊, 张晓青. 基于多粒度表征学习的加密恶意流量检测[J]. 计算机学报, 2023, 46(9): 1888-1899.
Gu Yonghao, Xu Hao, Zhang Xiaoqing. Multi-granularity representation learning for encrypted malicious traffic detection[J]. Chinese Journal of Computers, 2023, 46(9): 1888-1899. (in Chinese)

[18] 李锐光, 段鹏宇, 沈蒙, 等. 基于随机森林的物联网设备流量分类算法[J]. 北京航空航天大学学报, 2022, 48(2): 233-239.
Li Ruiguang, Duan Pengyu, Shen Meng, et al. Traffic classification algorithm of Internet of things devices based on random forest[J]. Journal of Beijing University of Aeronautics and Astronautics, 2022, 48(2): 233-239. (in Chinese)

[19] Hao Song, Fu Wentao, Chen Xuanze, et al. Network anomaly traffic detection via multi-view feature fusion[PP/OL]. V2. arXiv (2024-09-12)[2026-02-23]. https://arxiv.org/abs/2409.08020.

[20] Duan Chenxin, Zhang Shize, Yang Jiahai, et al. PIN-

BALL: Universal and robust signature extraction for smart home devices[C]//Proceedings of 2021 IFIP/IEEE International Symposium on Integrated Network Management. Piscataway: IEEE, 2021: 1-9.

[21] Dong Shuaike, Li Zhou, Tang Di, et al. Your smart home can't keep a secret: Towards automated fingerprinting of IoT traffic[C]//Proceedings of the 15th ACM Asia Conference on Computer and Communications Security. New York: ACM, 2020: 47-59.

[22] Sun Yizhen, Fu Shupo, Zhang Shigeng, et al. Accurate IoT device identification from merely packet length[C]//Proceedings of the 16th International Conference on Mobility, Sensing and Networking. Piscataway: IEEE, 2020: 774-778.

[23] Trimananda R, Varmarken J, Markopoulou A, et al. Packet-level signatures for smart home devices[C]//Proceedings of the Network and Distributed System Security Symposium. Reston: Internet Society, 2020.

[24] Pinheiro A J, de M. Bezerra J, Burgardt C A P,et al. Identifying IoT devices and events based on packet length from encrypted traffic[J]. Computer Communications, 2019, 144: 8-17.

[25] 李小龙, 李曦, 刘洋, 等. 基于延迟时空依赖的非平稳时间序列交通流量预测模型[J]. 电子学报, 2025, 53(11): 4035-4050.
Li Xiaolong, Li Xi, Liu Yang, et al. Non-stationary time series traffic flow forecasting model based on delayed spatio-temporal dependencies[J]. Acta Electronica Sinica, 2025, 53(11): 4035-4050. (in Chinese)

[26] Liu Xiangyu, Han Yi, Du Yanhui. IoT device identification using directional packet length sequences and 1D-CNN[J]. Sensors, 2022, 22(21): 8337.

[27] Oh S, Lee M, Lee H, et al. AppSniffer: Towards robust mobile app fingerprinting against VPN[C]//Proceedings of the ACM Web Conference. New York: ACM, 2023: 2318-2328.

[28] 孙学良, 黄安欣, 罗夏朴, 等. 针对Tor的网页指纹识别研究综述[J]. 计算机研究与发展, 2021, 58(8): 1773-1788.
Sun Xueliang, Huang Anxin, Luo Xiapu, et al. Webpage fingerprinting identification on Tor: A survey[J]. Journal of Computer Research and Development, 2021, 58(8): 1773-1788. (in Chinese)

[29] Williams M, Morales R, Johnson K, et al. Entropy-based network traffic analysis for efficient ransomware detection[EB/OL]. (2024-10-08)[2026-02-24]. https://doi.org/10.36227/techrxiv.172840776.66718131/v1.

[30] NetFlow analyzer[EB/OL]. [2026-02-23]. https://www.manageengine.com/products/netflow/.

[31] sFlow analyzer[EB/OL]. [2026-02-23]. https://www.manageengine.com/products/netflow/sflow-analyzer.html

[32] Vollmer M. Newton's law of cooling revisited[J]. European Journal of Physics, 2009, 30(5): 1063-1084.

[33] Manias D M, Chouman A L, Shami A. Model drift in dynamic networks[J]. IEEE Communications Magazine, 2023, 61(10): 78-84.

[34] Pashamokhtari A, Okui N, Nakahara M, et al. Dynamic inference from IoT traffic flows under concept drifts in residential ISP networks[J]. IEEE Internet of Things Journal, 2023, 10(17): 15761-15773.

[35] Sivanathan A, Gharakheili H H, Loi F, et al. Classifying IoT devices in smart environments using network traffic characteristics[J]. IEEE Transactions on Mobile Computing, 2019, 18(8): 1745-1759.

[36] Shahid M R, Blanc G, Zhang Zonghua, et al. IoT devices recognition through network traffic analysis[C]//Proceedings of 2018 IEEE International Conference on Big Data. Piscataway: IEEE, 2018: 5187-5192.

[37] Wang Tao, Cai Xiang, Nithyanand R, et al. Effective attacks and provable defenses for website fingerprinting[C]//Proceedings of the 23rd USENIX Security Symposium. Berkeley: USENIX Association, 2014: 143-157.

[38] Rimmer V, Preuveneers D, Juarez M, et al. Automated website fingerprinting through deep learning[C]//Proceedings of the 25th Annual Network and Distributed System Security Symposium. Reston: Internet Society, 2018: 1-15.

[39] Kolcun R, Popescu D A, Safronov V, et al. Revisiting IoT device identification[C]//Proceedings of the 5th Network Traffic Measurement and Analysis Conference. TMA, 2021.

[40] Taylor V F, Spolaor R, Conti M, et al. AppScanner: Automatic fingerprinting of smartphone apps from encrypted network traffic[C]//Proceedings of 2016 IEEE European Symposium on Security and Privacy. Piscataway: IEEE, 2016: 439-454.

[41] Aceto G, Ciuonzo D, Montieri A, et al. Traffic classification of mobile apps through multi-classification[C]//Proceedings of 2017 IEEE Global Communications Conference. Piscataway: IEEE, 2017: 8254059.

[42] Hayes J, Danezis G. k-fingerprinting: A robust scalable website fingerprinting technique[C]//Proceedings of the 25th USENIX Conference on Security Symposium. Berkeley: USENIX Association, 2016: 1187-1203.

[43] Sirinam P, Imani M, Juarez M, et al. Deep fingerprinting: Undermining website fingerprinting defenses with deep learning[C]//Proceedings of the 2018 ACM SIGSAC Conference on Computer and Communications Security. New York: ACM, 2018: 1928-1943.

[44] Shen Meng, Ji Kexin, Gao Zhenbo, et al. Subverting website fingerprinting defenses with robust traffic representation[C]//Proceedings of the 32nd USENIX Conference on Security Symposium. Berkeley: USENIX Association, 2023: 607-624.

作者简介

卞华峰　男,1997年8月出生于江苏省淮安市。现为西安交通大学计算机科学与技术学院博士研究生。主要研究方向为流量分析、无线安全。
E-mail: t1071680825@stu.xjtu.edu.cn

郑纪成　男,2001年8月出生于河北省承德市。现为西安交通大学计算机科学与技术学院博士研究生。主要研究方向为流量分析。
E-mail: 3123151037@stu.xjtu.edu.cn

马小博　男,1983年9月出生于陕西省宝鸡市。现为西安交通大学电信学部教授、博士生导师。主要研究方向为网络安全与隐私、网络测量。中国电子学会会员编号:E190036656M。
E-mail: xma.cs@xjtu.edu.cn

刘宏桃　男,1997年6月出生于山西省吕梁市。2024年毕业于西安交通大学软件学院,获工学硕士学位。主要研究方向为网络安全。
E-mail: lht974868031@stu.xjtu.edu.cn

管晓宏　男,1955年11月出生于四川省泸州市。现为西安交通大学电信学部教授、博士生导师,中国科学院院士。主要研究方向为复杂网络化系统的经济性与安全性,电力、能源、制造系统的优化,网络空间安全,信息物理融合系统等。中国电子学会会员编号:E190184944M。
E-mail: xhguan@xjtu.edu.cn