基于检索增强的少样本加密流量分类方法研究

摘要

随着网络加密技术的广泛应用，加密恶意流量分类已成为网络安全领域的重要研究方向 [1–3]。然而，在实际网络环境中，恶意样本标注稀缺、标注成本高昂，导致现有方法在少样本条件下面临明显性能瓶颈。针对这一问题，本文提出一种基于检索增强的轻量级加密流量分类模型 **RAC-ET**（Retrieval-Augmented Classification for Encrypted Traffic）。该方法首先将原始流量载荷字节转换为 token 序列，并利用预训练 ET-BERT 提取深层语义特征；随后构建基于向量索引的样本知识库，用于存储历史样本的高维表示；在推理阶段，通过高效向量检索获取与查询样本语义最相近的 Top-K 个近邻样本；最后，采用交叉注意力机制融合查询特征与检索特征，实现准确分类。在三个基准数据集上的实验结果表明：在 CICIDS2017 数据集上，RAC-ET 在少量标注样本场景下相较基线模型 ET-BERT 的平均准确率提升 6.72%，其中 10-shot 设定下取得最大提升 $+11.22\%$（$70.31\% \rightarrow 81.53\%$）；在 IoT-23 数据集的少样本场景下，准确率平均提升 $+20.09\%$，其中 3-shot 设定提升高达 $+28.15\%$（$20.43\% \rightarrow 48.57\%$）；在数据充足的 CSTNET-TLS 1.3 数据集（120 类）上，本方法在全量数据场景下的准确率达到 91.75%，较基线提升 3.0%。

关键词：加密流量分类；少样本学习；检索增强分类；预训练模型；网络安全

1 引言

随着 5G 通信技术的普及和数字基础设施不断完善，网络流量规模持续增长，网络攻击活动也呈现高频化与智能化趋势。Imperva 发布的《2025 Bad Bot Report》与美国联邦调查局（FBI）在 2026 年发布的有关加密货币和 AI 欺诈的公开通报表明，恶意自动化流量占比持续处于高位，基于生成式 AI 的诈骗活动在近两年显著增长；与此同时，网络犯罪带来的直接与间接经济损失不断扩大。相关公开资料显示，恶意机器人流量已占全网流量约 37%，而美国受害者因加密货币相关和 AI 辅助诈骗造成的损失已达数十亿美元量级 [4, 5]。对于多数攻击链路而言，恶意流量传输是实现渗透、横向移动与数据外泄的关键载体，因此高精度恶意流量检测已成为网络防御体系的基础能力。

然而，攻击者广泛使用 TLS/SSL 等加密协议隐藏通信语义，使依赖明文内容检测的传统方法（如 DPI）显著受限。尤其在 IoT 设备规模快速扩张的背景下，终端异构性与攻击面同步扩大，加密流量的高隐蔽性、强动态性与类别不均衡问题进一步加剧，给恶意流量识别带来持续挑战。已有研究指出，恶意软件传播正显著依赖 HTTPS、TLS 1.3 等加密连接 [2, 6]，这使传统依赖明文特征的深度包检测（DPI）手段逐渐失效，如何在海量且高隐蔽性的加密流量中精准识别恶意行为，已成为保障数字资产安全亟待解决的关键问题。

针对这一挑战，基于深度学习的方法凭借较强的自动特征提取与模式识别能力，已成为当前加密流量检测的主流技术路径。现有方法大致可分为三类。第一类是基于卷积神经网络（CNN）的方法。这类方法通常将原始流量字节流视为一维信号，或将其映射为二维表示后进行处理。代表性工作中，Wang 等人提出基于 1D-CNN 的端到端加密流量分类框架，可直接从原始流量中提取局部空间特征 [7]；Lotfollahi 等人进一步提出 DeepPacket 方法，利用 CNN 与自编码器联合实现加密流量的应用识别与流量分类 [1]。然而，CNN 类方法主要关注局部模式，往往难以充分建模数据包之间的长距离时序依赖，且固定输入长度的截断操作容易造成关键信息丢失。第二类是基于循环神经网络

（RNN/LSTM）的时序建模方法。该类方法将网络流量视为时间序列，并利用记忆机制捕捉上下文依赖；Rezaei 与 Liu 的代表性研究系统梳理并验证了 RNN/LSTM/GRU 路线在加密流量分类中的有效性 [2]；Liu 等人提出 FS-Net，通过编码器—解码器架构对流序列进行建模与重构，有效挖掘了加密流量中的深层时序特征 [10]。尽管此类方法在处理变长序列方面具有一定优势，但 RNN 架构并行效率较低，面对长序列时仍可能受梯度衰减影响，从而限制其对长流特征的稳定建模。第三类是基于预训练模型（Pre-trained Models）的方法。该类方法通常先在大规模无标注流量上进行自监督预训练，学习通用流量语义表示，再迁移到下游分类任务进行有监督微调，以提升特征表达能力。以 ET-BERT 为代表的工作借鉴 NLP 领域 BERT 架构，利用 Transformer 编码器捕捉长距离依赖关系，并通过掩码语言建模学习上下文感知的动态嵌入，在数据充足场景下取得了优异性能 [3]。然而，尽管预训练阶段无需标注数据，下游微调阶段仍高度依赖充足且分布均衡的标注样本；在 Few-Shot 场景中，由于监督信号不足，模型容易过拟合，性能显著下降。

尽管上述方法在特定场景下取得了一定进展，但在实际应用中仍面临三个关键挑战：（1）标注数据有限导致模型性能受限。在海量网络流量中，恶意样本占比较低，且标注依赖较强的专业知识，人工成本高昂；同时新型恶意软件变种不断涌现，但每类往往仅有少量标注样本，导致现有深度模型，尤其是 ET-BERT 等预训练模型，在微调阶段性能明显受限。（2）少样本条件下泛化能力不足。该问题主要体现在参数化判别模型（如 CNN、RNN/LSTM 及 ET-BERT 微调模型）上：模型知识高度内化于参数，面对隐蔽性强、行为模式演化快的新型攻击时，难以充分利用有限样本中的上下文信息，决策边界容易波动。（3）模型更新与在线适配成本较高。该问题主要体现在需要频繁重训的监督式深度分类器上，尤其是大参数量预训练模型：当攻击模式持续演化时，往往需要重新微调大量参数或周期性全量训练，难以满足实际部署中低时延和快速迭代的要求。为此，本文提出一种基于检索增强的轻量级加密流量分类模型 **RAC-ET**（Retrieval-Augmented Classification for Encrypted Traffic）。该方法首先将原始流量数据包载荷字节转换为 token 序列，并利用预训练 ET-BERT 作为特征提取器获得深层语义表示；随后构建基于向量索引的样本知识库，以存储历史样本的高维特征；最后在推理阶段检索与查询样本语义最相近的 Top-K 近邻样本，并通过交叉注意力机制融合查询特征与检索特征，实现准确分类。通过检索并融合相似样本信息，该方法有效扩展了分类决策可利用的上下文，从而显著提升了少量标注样本场景下的分类性能。

本文主要贡献如下：（1）提出一种基于检索增强的加密流量分类框架，将预训练流量表示与相似样本检索机制有机结合，通过引入外部知识库作为非参数记忆，缓解预训练模型在少样本场景下的性能退化；（2）设计基于交叉注意力的特征融合机制，在保持模型轻量化（仅训练少量参数）的同时，实现查询特征与近邻特征的自适应整合；（3）构建覆盖少样本与全样本场景的系统实验评估方案，并结合消融实验分析检索规模 K、残差连接和注意力层等关键设计对模型性能的影响。

2 相关工作

2.1 加密流量分类

加密流量是指通过 SSL/TLS、SSH 等协议封装，使原始载荷呈现高熵乱码特征的数据流。随着全球网络隐私保护的增强，加密流量在互联网总流量中的占比已超过 90

传统机器学习方法主要依赖人工设计的统计特征，即由专家根据网络协议与流行为先验经验手工构造可区分特征（如流持续时间、包长分布、方向序列、到达间隔、TLS 握手元数据等），再输入随机森林、支持向量机等分类器进行判别。代表性工作中，Anderson 等人在 2018 年通过提取 TLS 握手阶段的未加密元数据（如密码套件、扩展字段）并结合机器学习模型识别恶意流量 [6]；Draper-Gil

等人则利用流持续时间与包长统计等时间相关特征刻画 VPN 流量 [8]。这类方法具有可解释性较强、实现门槛较低的优点，但高度依赖专家经验进行特征工程，难以适应特征日益隐蔽且持续演化的复杂攻击。需要说明的是，上述文献虽早于近五年，但属于加密流量分类的重要奠基工作。

深度学习方法凭借端到端自动特征提取能力逐渐成为主流。早期工作中，Wang 等人提出基于 1D-CNN 的端到端框架，直接以原始流量字节为输入，避免了繁琐的人工特征筛选 [7]；Shapira 等人提出 FlowPic，将流量包大小与到达时间映射为二维图像并利用 2D-CNN 完成分类 [9]；Liu 等人提出 FS-Net，通过流序列建模与重构机制挖掘深层时序特征，在加密流量分类任务中取得了较好效果 [10]。近五年研究进一步聚焦“预训练 + 轻量化 + 泛化能力”：Lin 等人提出 ET-BERT，将网络数据包视为文本并通过掩码语言建模学习上下文表征 [3]；随后，针对头部信息编码与跨场景泛化，BERT-Packet-Header 与 MetaRockETC 等方法在公开数据集上进一步提升了加密流量分类性能 [11,12]。此外，近期综述与实证研究也指出，深度模型在跨域部署和类别演化场景下仍面临鲁棒性与可迁移性挑战 [13,14]。综上，现有方法已从传统机器学习演进到深度学习，并进一步发展到预训练表征学习范式，但“少样本条件下的稳定泛化”仍是该领域的核心挑战。

2.2 少样本学习

少样本学习（Few-Shot Learning, FSL）关注在极少标注样本条件下实现有效分类，其核心在于提升模型对新类别、新任务的快速适应能力。在网络安全场景中，由于恶意流量标注依赖专家经验、样本获取成本高且攻击类型演化迅速，FSL 因而具有较强的应用意义。

现有 FSL 方法主要包括两条技术路线。第一类是元学习方法，即通过大量辅助任务训练模型” 如何学习”。Vinyals 等人提出 Matching Networks，通过注意力机制在支撑集与查询样本之间建立端到端匹配关系，使模型能够依据样本间相似性直接完成判别 [16]；Snell 等人提出 Prototypical Networks，将每一类样本映射为特征空间中的原型向量，并利用查询样本到各类原型的距离完成分类，结构简洁且具有较好的稳定性 [15]；Finn 等人提出 MAML，通过双层优化学习一组易于迁移的初始化参数，使模型只需少量梯度更新即可适应新任务 [17]。这类方法强调任务级知识迁移，在标准 few-shot 基准上取得了较好效果。

第二类是迁移学习与参数高效微调方法，其基本思想是先利用大规模数据获得通用表征，再通过更新少量附加参数完成目标任务适配。Houlsby 等人提出 Adapter，在预训练模型层间插入轻量瓶颈模块，仅训练新增参数即可实现跨任务迁移 [18]；Hu 等人提出 LoRA，将权重更新约束为低秩分解形式，在基本不增加推理开销的前提下显著降低微调成本 [19]。相较于全参数微调，这类方法更适合样本有限、计算资源受限的应用环境。

在加密流量分类领域，Yang 等人提出 MetaMRE，将元学习机制与表征增强策略结合，通过任务级自适应特征变换提升模型在小样本流量分类中的泛化能力，验证了 FSL 思想在网络流量场景中的可行性 [20]。

然而，上述方法在加密恶意流量检测任务中仍存在若干局限。首先，元学习方法通常依赖大量与目标任务分布相近的辅助任务进行训练，但真实网络环境中的流量类别演化快、分布漂移明显，导致离线构造的训练任务难以充分覆盖实际攻击模式。其次，无论是 Matching Networks、Prototypical Networks 还是 MAML，这类方法大多侧重从当前支撑集内部提取判别依据；当每类样本极少且类内差异较大时，模型容易受到支撑样本代表性不足的影响。再次，Adapter、LoRA 等参数高效微调方法虽然降低了训练成本，但其知识仍主要保存在模型参数中，对新增攻击类型和历史相似样本的利用能力有限，难以像显式记忆机制那样动态吸收外部信息。因而，仅依赖元学习或轻量微调，仍

不足以稳定解决少样本加密流量分类中的泛化与在线适应问题，这也为引入检索增强的外部样本记忆提供了动机。

2.3 检索增强方法

检索增强（Retrieval-Augmented, RA）技术旨在通过引入外部知识库，弥补深度学习模型在长尾知识记忆与实时数据更新方面的局限性。其核心逻辑是将模型的参数化知识（Parametric Knowledge）与外部知识库中的非参数化知识（Non-Parametric Knowledge）进行协同融合。按任务目标划分，现有研究主要沿两条路径演进：检索增强生成（RAG）与检索增强分类（RAC）。

检索增强生成（Retrieval-Augmented Generation, RAG）是当前 NLP 领域的代表范式。典型 RAG 系统由神经检索器与生成模型组成：在推理阶段，模型先根据查询从外部语料库检索相关片段，再将其作为上下文引导生成器输出结果 [21]。围绕“如何提升检索质量”，后续工作进一步发展了以稠密向量检索为核心的双塔编码器方案（如 DPR），显著增强了知识密集任务中的召回能力与语义匹配质量 [22]；近年的 Self-RAG 进一步引入按需检索与自反思机制，以提升生成质量与事实一致性 [23]。然而，RAG 类方法通常依赖较大规模生成模型，推理链路长、计算开销高，难以直接满足网络流量检测场景对低时延与高吞吐的工程约束。

与生成任务不同，检索增强分类（Retrieval-Augmented Classification, RAC）的目标是检索相似样本来辅助判别。该方法可追溯到 kNN-LM 等非参数记忆增强方法，即在参数模型之外引入近邻检索以校准预测分布 [24]；近年来，相关研究进一步采用稠密向量空间中的 Top-K 近邻检索与特征融合机制，提升模型在长尾类别与低资源样本条件下的鲁棒性；例如在 Few-Shot 文本分类中，检索增强训练目标被用于稳定提升低资源场景性能 [25]，并在长文档分类中发展出兼顾效率与可解释性的检索增强判别框架 [26]。相较之下，在加密流量分类领域，现有研究仍主要集中于特征表示学习与参数化分类器优化，对于“利用外部相似样本为当前流量判别提供辅助上下文”的思路讨论相对较少，尤其在少样本条件下，如何将检索到的历史样本有效转化为稳定的分类依据，仍缺乏充分研究。

针对这一问题，本文借鉴 kNN-LM 的非参数记忆增强思路以及 RAC 范式中检索辅助判别的核心思想，提出面向加密流量少样本分类的改进方案：不同于依赖生成式大模型的路径，本文使用预训练 ET-BERT 构建流量语义样本知识库；在推理时通过高效向量检索获取与查询样本最相近的历史近邻样本；随后引入交叉注意力（Cross-Attention）机制自适应融合查询特征与检索特征。该设计避免了复杂文本生成过程，能够更充分地利用外部标注样本信息，在少样本条件下增强分类边界判别能力，并兼顾实际部署所需的效率与稳定性，实现了轻量级、可动态更新的检索增强加密流量分类。

需要强调的是，RAC-ET 与传统 k-最近邻分类器（kNN）存在本质区别：（1）kNN 直接以检索到的近邻标签投票作为最终预测，是一种*非参数判别方法*；而 RAC-ET 将检索到的近邻特征通过*可学习的交叉注意力模块*与查询特征自适应融合，再由参数化分类头输出预测，检索结果并非直接投票而是作为” 上下文增强” 参与融合决策。（2）kNN 对近邻质量高度敏感，一旦检索到噪声样本即直接影响分类结果；RAC-ET 的注意力机制能够学习抑制噪声近邻的权重，从消融实验（表 7）中可以看到，去除注意力机制后性能显著下降，证明了自适应融合对于利用检索信息的必要性。（3）kNN 没有训练阶段，无法对检索—融合过程进行端到端优化；RAC-ET 通过有监督损失联合优化交叉注意力、FFN 与分类头，使模型能够学习” 如何最优地利用检索结果”。

3 方法

本节围绕所提出的检索增强加密流量分类模型 **RAC-ET**（Retrieval-Augmented Classification for Encrypted Traffic）展开介绍。参考上传论文在“方法”章节中的组织方式，本文按照“整体框架—数据预处理—特征表示—检索增强—融合判别”的逻辑，对模型各组成模块及其协同机制进行说明。为避免符号混淆，下文使用 K_s 表示少样本设置中每类可用标注样本数（即 K_s-shot），使用 K_r 表示检索阶段返回的近邻数（Top-K_r）。

3.1 框架概述

RAC-ET 的整体架构如图 1 所示，主要包括数据预处理、特征提取、知识库构建与近邻检索、交叉注意力融合以及分类决策五个环节。与传统仅依赖参数化分类器的加密流量检测方法不同，RAC-ET 在保留预训练模型语义表征能力的基础上，引入外部样本记忆作为检索上下文，使模型在少样本条件下仍能够借助历史相似样本完成稳定判别。其核心组件可概括为：

- **特征提取器**：使用预训练的 ET-BERT [3] 将原始流量字节序列编码为稠密特征向量；
- **知识库**：存储可用流量样本特征表示的向量数据库；
- **检索模块**：通过高效相似性搜索从知识库中检索与查询样本最相关的近邻；
- **融合分类器**：利用交叉注意力机制融合查询特征与检索特征，并输出最终类别预测。

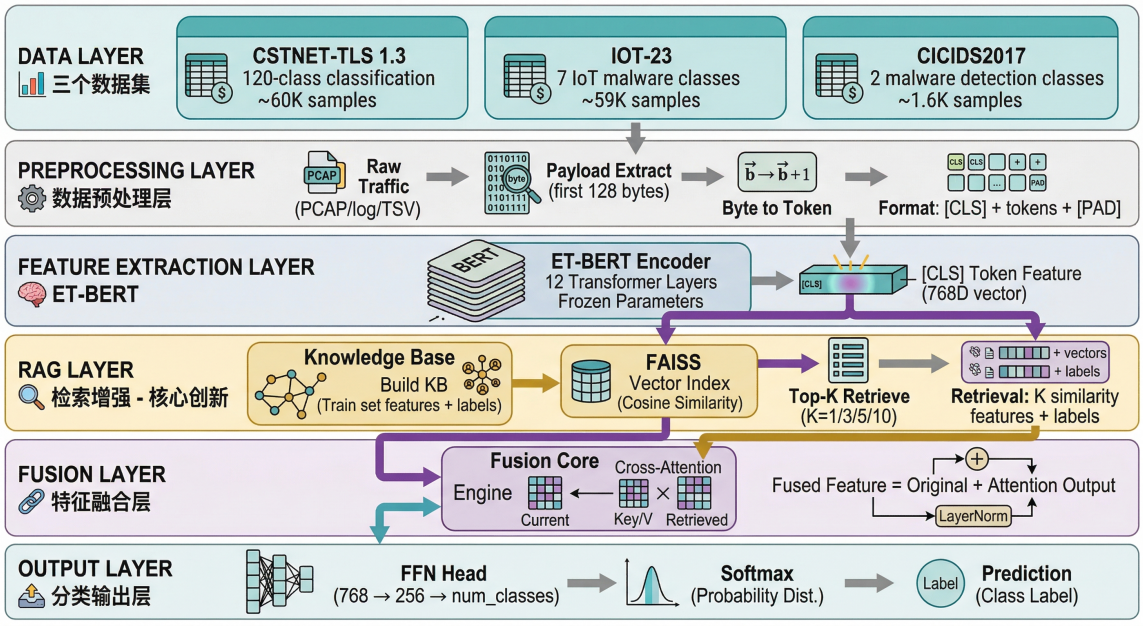

Figure 1: RAC-ET 方法整体架构示意图。输入流量经 ET-BERT 编码为查询特征，再从知识库中检索 Top-K_r 相似样本，并通过交叉注意力与前馈网络进行融合，最后由分类头输出预测类别。

给定输入流量样本 x，RAC-ET 的整体流程如下：

1. 使用冻结的 ET-BERT 编码器提取查询特征：

$$\mathbf{h}_q = f(x). \tag{1}$$

2. 从知识库中检索 Top-K_r 个相似样本：

$$\mathcal{R}(x) = \{(x_1, y_1), \ldots, (x_{K_r}, y_{K_r})\}. \tag{2}$$

3. 提取检索样本的特征表示：

$$\{\mathbf{h}_1, \mathbf{h}_2, \ldots, \mathbf{h}_{K_r}\}. \tag{3}$$

4. 通过交叉注意力模块融合查询特征与检索特征，得到增强表示 $\mathbf{h}_{\text{fused}}$；

5. 将融合表示输入分类头，输出最终预测结果。

3.2 数据预处理

遵循 ET-BERT 的输入规范 [3]，本文首先将原始加密流量转换为适用于 Transformer 编码器处理的离散 token 序列。该过程对应模型训练前的数据标准化环节，其目标是尽可能保留有效载荷中的行为语义，同时统一输入长度与表示形式。整个预处理流程包含以下三个阶段。

阶段 1：数据包提取 对于每个网络流，首先从传输层（TCP/UDP）提取载荷字节。具体步骤如下：

1. 解析以太网帧并提取 IP 数据包；
2. 识别传输层协议类型；
3. 提取传输层头部之后的有效载荷字节；
4. 将载荷字节序列截断或补齐到固定长度 $L_0 = 128$。

阶段 2：字节到 Token 的转换 对载荷中的每个字节 $b_j \in \{0, \ldots, 255\}$，采用领域特定词表映射为 token t_j：

$$t_j = \begin{cases} b_j + 1, & b_j \in \{0, \ldots, 255\}, \\ 0, & \text{若该位置为填充。} \end{cases} \tag{4}$$

因此，词表大小为 257，其中 token 0 表示填充符，token $1 \sim 256$ 对应字节值 0x00–0xFF。

阶段 3：序列格式化 在 token 序列首部添加 [CLS] 标记，并对长度不足 L_0 的序列使用 [PAD] 填充。最终输入格式为

$$\text{[CLS]}, t_1, t_2, \ldots, t_n, \text{[PAD]}, \ldots, \text{[PAD]}. \tag{5}$$

因此，ET-BERT 的实际输入长度为

$$T = L_0 + 1 = 129. \tag{6}$$

3.3 基于 ET-BERT 的特征提取

在完成 token 化之后，本文采用 ET-BERT 作为骨干特征提取器，对流量序列进行统一编码。ET-BERT 是一个面向网络流量场景预训练的 BERT 模型，其通过掩码语言建模任务在大规模无标注流量上学习上下文依赖，因此能够较好地捕捉加密流量中的统计模式与结构信息。

给定输入序列 x，ET-BERT 为每个 token 生成上下文化表示：

$$\mathbf{H} = \text{ET-BERT}(x) \in \mathbb{R}^{T \times d}, \tag{7}$$

其中 T 表示输入序列长度，$d = 768$ 表示隐藏维度。

本文采用 [CLS] 位置对应的隐藏状态作为整个流量样本的全局表示，即

$$\mathbf{h} = \mathbf{H}_{[\text{CLS}]} \in \mathbb{R}^{d}. \tag{8}$$

在训练与推理过程中，ET-BERT 参数保持冻结，以充分保留预训练知识并降低微调开销。

3.4 知识库构建

知识库存储可用流量样本的特征表示及其标签信息，用于在推理阶段为查询样本提供外部上下文支持。对于每个可用样本 (x_i, y_i)，首先通过冻结的特征提取器获得其表示

$$\mathbf{h}_i = f(x_i). \tag{9}$$

随后，构建知识库

$$\mathcal{KB} = \{(\mathbf{h}_i, y_i) \mid i = 1, 2, \ldots, N\}. \tag{10}$$

本文使用 FAISS [28] 构建向量索引，以支持高效近邻检索。考虑到检索阶段采用余弦相似度，为了与 `IndexFlatIP` 的内积搜索保持一致，在索引写入与查询检索前均对特征向量进行 L_2 归一化。为表述简洁，后文仍将归一化后的向量记为 $\mathbf{h}$。

在少样本场景下，若任务为 C-way K_s-shot，则仅由标注数据构建的知识库规模为 $K_s \times C$。此外，若存在额外未标注样本或保留样本，也可以仅将其特征表示加入检索候选池，以提供更丰富的上下文信息；这类样本不参与监督损失计算。

对于更大规模的知识库，除精确检索外，还可以进一步采用 IVF、HNSW 等近似最近邻索引结构以提升检索效率。

3.5 检索模块

在获得查询样本的语义表示后，检索模块从知识库中选出与其最相似的 Top-K_r 个样本，作为当前分类任务的支撑实例。本文采用余弦相似度作为相似性度量：

$$s(\mathbf{h}_q, \mathbf{h}_i) = \frac{\mathbf{h}_q^{\top} \mathbf{h}_i}{\|\mathbf{h}_q\|_2 \, \|\mathbf{h}_i\|_2}. \tag{11}$$

由于特征已进行 L_2 归一化，上式可等价写为内积形式

$$s(\mathbf{h}_q, \mathbf{h}_i) = \mathbf{h}_q^\top \mathbf{h}_i. \tag{12}$$

因此，检索集合可表示为

$$\mathcal{R}(x) = \text{TopK}_{(\mathbf{h}_i, y_i) \in \mathcal{KB}}\, s(\mathbf{h}_q, \mathbf{h}_i). \tag{13}$$

最终得到的检索结果为

$$\mathcal{R}(x) = \{(\mathbf{h}_1, y_1), (\mathbf{h}_2, y_2), \ldots, (\mathbf{h}_{K_r}, y_{K_r})\}, \tag{14}$$

其中每个近邻样本均由其特征向量与标签组成。该过程本质上是在参数化表示空间中引入一层可动态更新的非参数记忆，有助于提升模型对长尾类别和少样本类别的判别稳定性。

3.6 交叉注意力融合

为有效利用检索到的上下文信息，本文设计交叉注意力融合模块，将查询特征与检索特征进行选择性聚合 [29]。与直接拼接或平均池化不同，该模块能够根据当前查询样本的语义需求，自适应评估不同近邻样本的重要性。设查询特征为 $\mathbf{h}_q \in \mathbb{R}^{1\times d}$，将 K_r 个检索特征堆叠为

$$\mathbf{H}_r = [\mathbf{h}_1; \mathbf{h}_2; \ldots; \mathbf{h}_{K_r}] \in \mathbb{R}^{K_r \times d}. \tag{15}$$

交叉注意力的计算过程为

$$\mathbf{Q} = \mathbf{h}_q \mathbf{W}_Q, \tag{16}$$

$$\mathbf{K} = \mathbf{H}_r \mathbf{W}_K, \tag{17}$$

$$\mathbf{V} = \mathbf{H}_r \mathbf{W}_V, \tag{18}$$

其中 $\mathbf{W}_Q, \mathbf{W}_K, \mathbf{W}_V \in \mathbb{R}^{d\times d_k}$ 为可学习投影矩阵。随后，注意力输出为

$$\mathbf{z} = \text{softmax}\left(\frac{\mathbf{Q}\mathbf{K}^\top}{\sqrt{d_k}}\right)\mathbf{V}. \tag{19}$$

为保证与原始查询特征维度一致，引入输出投影矩阵 $\mathbf{W}_O \in \mathbb{R}^{d_k\times d}$，并通过残差连接与层归一化得到

$$\mathbf{h}_{\text{attn}} = \text{LayerNorm}\left(\mathbf{h}_q + \mathbf{z}\mathbf{W}_O\right). \tag{20}$$

在此基础上，进一步使用前馈网络（FFN）增强特征表达能力：

$$\text{FFN}(\mathbf{x}) = \text{ReLU}(\mathbf{x}\mathbf{W}_1 + \mathbf{b}_1)\mathbf{W}_2 + \mathbf{b}_2, \tag{21}$$

$$\mathbf{h}_{\text{fused}} = \text{LayerNorm}\left(\mathbf{h}_{\text{attn}} + \text{FFN}(\mathbf{h}_{\text{attn}})\right). \tag{22}$$

上述机制能够根据查询样本的语义表示，自适应地为不同检索样本分配权重，从而突出与当前分类决策最相关的上下文信息，并抑制无关或噪声近邻带来的干扰。

3.7 分类头

在得到融合表示后，本文使用轻量级线性分类头完成最终判别。融合后的特征 $\mathbf{h}_{\text{fused}}$ 经过线性分类层并通过 softmax 得到类别概率分布：

$$\mathbf{p} = \text{softmax}\left(\mathbf{h}_{\text{fused}}\mathbf{W}_c + \mathbf{b}_c\right), \tag{23}$$

其中 $\mathbf{W}_c \in \mathbb{R}^{d\times C}$、$\mathbf{b}_c \in \mathbb{R}^C$，$C$ 表示类别数。

3.8 训练目标

本文采用交叉熵损失对模型进行训练，以最小化预测类别分布与真实标签分布之间的差异。对于包含 N 个样本的训练集，其损失函数定义为

$$\mathcal{L}_{\text{ce}} = -\frac{1}{N}\sum_{i=1}^{N}\sum_{c=1}^{C} y_{i,c}\log p_{i,c}, \tag{24}$$

其中 $y_{i,c}$ 为样本 i 在类别 c 上的独热标签，$p_{i,c}$ 为模型预测其属于类别 c 的概率。

由于 ET-BERT 骨干网络保持冻结，训练过程中仅更新交叉注意力模块、FFN 以及分类头参数，从而显著降低优化开销，并使模型更适合少样本条件下的快速适配。

3.9 参数效率分析

通过冻结 ET-BERT 骨干网络，RAC-ET 仅需训练少量附加参数即可完成下游分类任务。表 1 给出了各模块的参数分布情况。

Table 1: RAC-ET 框架的参数效率分析。

组件	参数量	是否可训练
ET-BERT 骨干	132.2M	否（冻结）
交叉注意力模块	2.36M	是
FFN 层	2.36M	是
分类头	0.35M	是
总计	137.3M	5.07M（3.69%）

可以看出，本文方法仅训练总参数量的 3.69%，在显著降低计算成本的同时，仍能够充分利用预训练模型的表征能力与检索增强带来的上下文信息。

3.10 整体算法

算法 1 给出 RAC-ET 的完整训练与推理流程伪代码。

Algorithm 1 RAC-ET: Retrieval-Augmented Classification for Encrypted Traffic

Require: 预训练编码器 $\mathcal{E}$（冻结）；训练集 $\mathcal{D}_{\text{train}}$；测试集 $\mathcal{D}_{\text{test}}$；每类样本数 K_s；检索规模 K_r
Ensure: 测试集预测标签 $\hat{\mathbf{y}}$

// **阶段 1：特征提取与知识库构建**
for $(\mathbf{x}_i, y_i) \in \mathcal{D}_{\text{train}}$ **do**
  $\mathbf{h}_i \leftarrow \mathcal{E}(\mathbf{x}_i)$ ▷ 提取 768 维特征
  $\mathbf{h}_i \leftarrow \mathbf{h}_i / \|\mathbf{h}_i\|_2$ ▷ L_2 归一化
end for
构建 FAISS 索引 $\mathcal{I} \leftarrow \text{IndexFlatIP}(\{\mathbf{h}_i\})$
// **阶段 2：融合模块训练**
for epoch $= 1, \ldots, E$ **do**
  for mini-batch $\mathcal{B} \subset \mathcal{D}_{\text{train}}$ **do**
    $\mathbf{h}_q \leftarrow \mathcal{E}(\mathbf{x}_q)$ ▷ 查询特征
    $\mathbf{h}_{r_1}, \ldots, \mathbf{h}_{r_{K_r}} \leftarrow \text{FAISS-Search}(\mathcal{I}, \mathbf{h}_q, K_r)$
    $\mathbf{h}_{\text{fused}} \leftarrow \text{CrossAttn}(\mathbf{h}_q, [\mathbf{h}_{r_1}; \ldots; \mathbf{h}_{r_{K_r}}])$
    $\mathbf{h}_{\text{out}} \leftarrow \text{FFN}(\mathbf{h}_{\text{fused}}) + \mathbf{h}_q$ ▷ 残差连接
    $\hat{y}_q \leftarrow \text{softmax}(\mathbf{W}_c \cdot \mathbf{h}_{\text{out}})$
    反向传播更新 CrossAttn, FFN, $\mathbf{W}_c$ 参数
  end for
end for
// **阶段 3：推理**
for $\mathbf{x}_t \in \mathcal{D}_{\text{test}}$ **do**
  $\mathbf{h}_t \leftarrow \mathcal{E}(\mathbf{x}_t) / \|\mathcal{E}(\mathbf{x}_t)\|_2$
  $\mathbf{h}_{r_1}, \ldots, \mathbf{h}_{r_{K_r}} \leftarrow \text{FAISS-Search}(\mathcal{I}, \mathbf{h}_t, K_r)$
  $\hat{y}_t \leftarrow \arg\max \ \text{softmax}(\mathbf{W}_c \cdot (\text{FFN}(\text{CrossAttn}(\mathbf{h}_t, [\mathbf{h}_r])) + \mathbf{h}_t))$
end for
return $\hat{\mathbf{y}}$

4 实验

本节在 CSTNET-TLS 1.3（应用识别）、IoT-23（IoT 恶意软件识别）与 CICIDS2017（网络入侵检测）三个基准数据集上对 RAC-ET 进行系统评估。围绕方法的核心问题——检索增强能否在少样本加密流量分类中稳定改善预训练模型的表现——我们依次给出实验设置（4.1）、主结果对比（4.2）、K 值与组件消融（4.3）、效率分析（4.4）、CICIDS2017 深入分析（4.5）以及讨论（4.6）。

4.1 实验设置

数据集 本文选取三个公开数据集以覆盖不同的分类粒度、数据稀缺程度与输入模态。

CSTNET-TLS 1.3：面向应用分类的大规模 TLS 1.3 加密流量数据集，共 120 个类别，代表类别多、数据充足的应用识别场景。本文沿用 ET-BERT 官方预处理流程，训练集约 50,000 条流量，测试集约 10,000 条。

IoT-23：面向 IoT 恶意软件家族识别的公开数据集，原始数据包含 23 个攻击场景，本文按官方标注将其映射为 {Benign, Okiru, Torii, C&C, DDoS, Attack, PortScan} 共 7 个类别；预处理后训练集约 1.38×10^5 条、测试集约 3.4×10^4 条，代表类别较少但跨场景、类别极不均衡的恶意流量检测场景。

CICIDS2017：面向网络入侵检测的经典公开数据集 [30]，包含 5 天的网络流量采集，使用 CICFlowMeter 提取的 78 维流统计特征作为输入。本文将原始 15 类攻击合并为语义相近的 7 类（Benign, Brute_Force, DoS, DDoS, Botnet, PortScan, Web_Attack），过滤样本数不足 50 的稀有类别（Heartbleed, Infiltration）；训练集约 35,000 条、测试集约 8,800 条，代表流特征输入、跨模态泛化验证的场景。三个数据集的整体统计信息如表 2 所示。

Table 2: 实验所用数据集统计信息。

数据集	类别数	训练样本	测试样本	特征类型
CSTNET-TLS 1.3	120	∼50,000	∼10,000	字节载荷
IoT-23	7	∼138,000	∼34,000	字节载荷
CICIDS2017	7	∼35,000	∼8,800	流特征

少样本设置 对每个数据集的每一类别，从训练集中采样 $K_s \in \{1, 3, 5, 10, 20\}$ 个样本构成少样本训练集，其余训练样本作为候选池加入知识库但不参与监督损失。少样本训练集同时用于训练融合模块（交叉注意力 + FFN + 分类头）并构造初始知识库。为降低采样方差，少样本实验均以 3 个不同随机种子（42/123/456）重复，报告准确率（Accuracy, Acc）与宏平均 F1（Macro-F1）的均值。

骨干与检索配置 特征提取采用 ET-BERT [3] 官方开源模型，隐藏维度 $d = 768$，在 CSTNET 上使用对应的 `finetuned_model.bin` 与 `encryptd_vocab.txt`，在 IoT-23 上使用对应的 `finetuned_model.bin` 与相同的字节级词表；ET-BERT 参数在训练与推理阶段保持冻结。输入序列按第 3.2 节所述的流程统一预处理为长度 $T = 129$ 的 token 序列。知识库使用 FAISS [28] 的 `IndexFlatIP` 索引，特征均经 L_2 归一化后写入，使用内积等价实现余弦相似度检索。默认检索规模 $K_r = 1$，具体取值在 4.3 节做消融。

CICIDS2017 特征编码 为验证 RAC-ET 框架对不同输入模态的泛化能力，在 CICIDS2017 上采用 CICFlowMeter 提取的 78 维流统计特征（包括流持续时间、包长分布、到达间隔等）作为输入。为将流特征投射到与 ET-BERT 相同的 768 维表征空间，设计了一个轻量级的流特征编码器（两层全连接网络，中间维度 384，LayerNorm + GELU 激活），并通过自监督对比学习（SimCLR-style，温度系数 $\tau = 0.1$）在全量未标注流特征上预训练 50 个 epoch，不使用任何类别标签。该编码器的作用等价于 ET-BERT 在 CSTNET/IoT-23 上的角色：提供冻结的通用表征，使后续检索与分类模块在完全相同的 768 维空间上运行。实验中合并语义相近的攻击子类别为 7 类（Benign, Brute_Force, DoS, DDoS, Botnet, PortScan, Web_Attack），过滤样本数不足 50 的稀有类（Heartbleed, Infiltration）。

融合与训练 交叉注意力模块与 FFN 均采用单层结构，隐藏维度 $d_k = 768$、注意力头数为 8。优化器为 Adam，学习率 2×10^{-5}，线性预热 10% 步数，在少样本设置下训练 30 个 epoch（早停容忍 5 个 epoch），批大小为 32。在全量 CSTNET 训练时，batch 大小为 64，训练 10 个 epoch。

基线 实验对比以下基线：

- **ET-BERT** 微调 [3]：在 ET-BERT 的 `[CLS]` 表征上接线性分类头并整体微调；与 RAC-ET 共享相同骨干与输入，是本文方法最直接的对照基线；
- **随机森林**、**XGBoost**：以 ET-BERT 全局特征为输入的传统机器学习分类器，用于对比传统判别器的上限；
- 少样本基线中附加**原型网络 (Prototypical Network)** [15] 与**匹配网络 (Matching Network)** [16]。

实现与硬件 所有实验在单卡 NVIDIA RTX 3090 Ti (24 GB) 上进行，基于 PyTorch 实现；FAISS 使用 CPU 版本构建索引；评估指标为 Accuracy 与 Macro-F1。

4.2 主结果

少样本性能 表 3（Accuracy）与表 4（Macro-F1）、图 2、图 3 与图 4 汇总了三个数据集在不同 K_s 设定下的少样本结果（均值，$K_r = 1$ 或取最优 K_r）。总体来看，RAC-ET 在三个数据集的所有设置上均不劣于基线，并在 IoT-23 与 CICIDS2017 上带来显著提升。

对上述结果，我们观察到以下现象：

1. **语义覆盖充分的场景下保持稳定**。由于 ET-BERT 本身即在 CSTNET 同源语料上完成掩码建模预训练，其 `[CLS]` 表征在该任务上已非常紧凑，仅 1-shot 亦可达到 95.44% 的高准确率；在此基础上 RAC-ET 借助检索上下文带来 $+0.00\% \sim +0.30\%$ 的一致提升而未引入副作用，表明引入非参数记忆并不会损伤原有判别边界。
2. **语义覆盖不足的场景下显著提升**。IoT-23 的恶意流量分布与 ET-BERT 预训练语料差异较大，仅凭 `[CLS]` 表征难以对 7 个家族形成稳定判别，ET-BERT 在 1-shot 下仅达到 24.14%；RAC-ET 通过检索与融合支撑样本特征，将 3-shot/5-shot/10-shot/20-shot 的准确率分别提升 $+28.15\%/+20.11\%/+26.23\%/+23.73\%$，平均增益约 $+20.09\%$。

Table 3: 少样本分类主结果（Accuracy，%）。ΔAcc 为 RAC-ET 相对 ET-BERT 基线的绝对提升。

数据集	设定	ET-BERT	RAC-ET (Ours)	ΔAcc
CSTNET-TLS 1.3	1-shot	95.44	**95.44**	+0.00
	5-shot	95.38	**95.60**	+0.22
	10-shot	95.52	**95.82**	+0.30
IoT-23	1-shot	24.14	**26.35**	+2.21
	3-shot	20.43	**48.57**	+28.15
	5-shot	31.25	**51.35**	+20.11
	10-shot	36.33	**62.56**	+26.23
	20-shot	44.91	**68.65**	+23.73
CICIDS2017	1-shot	35.24	**36.90**	+1.66
	5-shot	60.23	**67.52**	+7.29
	10-shot	70.31	**81.53**	+11.22

Table 4: 少样本分类主结果（Macro-F1，%）。ΔF1 为 RAC-ET 相对 ET-BERT 基线的绝对提升。

数据集	设定	ET-BERT	RAC-ET (Ours)	ΔF1
CSTNET-TLS 1.3	1-shot	95.44	**95.44**	+0.00
	5-shot	95.38	**95.60**	+0.22
	10-shot	95.52	**95.82**	+0.30
CICIDS2017	1-shot	36.31	**38.45**	+2.14
	5-shot	60.70	**66.30**	+5.60
	10-shot	70.74	**80.58**	+9.84

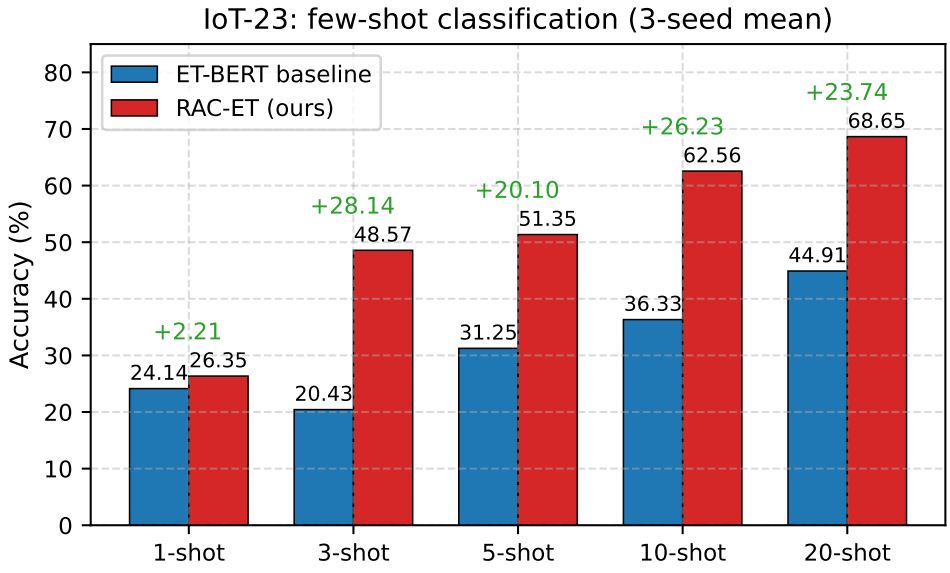

Figure 2: IoT-23 数据集上不同 K_s 下 RAC-ET 与 ET-BERT 基线的准确率对比（3 个种子均值）。绿色数值为 RAC-ET 相对基线的绝对提升。

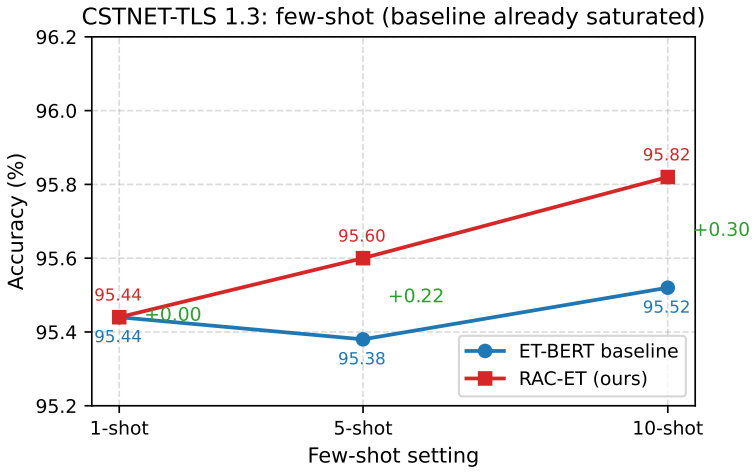

Figure 3: CSTNET-TLS 1.3 上少样本设定下的准确率对比。ET-BERT 基线已接近饱和，RAC-ET 仍取得 +0.00%～+0.30% 的一致提升且不引入退化。

3. **增益随监督数据的可分辨性而非数量单调变化**。1-shot 下仅有 7 条支撑样本，随机性较大且类簇稀疏，RAC-ET 的提升相对保守（+2.21%）；当 $K_s \geq 3$ 时，知识库中可利用的同类样本快速增多，提升跃升至 +20% 以上并趋于稳定。

4. **跨模态泛化：检索增强在流特征空间上同样有效**。在 CICIDS2017 上使用 78 维流统计特征替代字节载荷（图 4），RAC-ET 仍然展现了一致且显著的提升：1-shot +1.66%、5-shot +7.29%、10-shot +11.22%。该结果表明 RAC-ET 的核心优势——通过检索近邻扩展少样本决策上下文——并不依赖特定的骨干编码器，在自监督预训练的流特征编码器上同样成立。值得注意的是，10-shot 下的绝对提升达 11.22%（从 70.31% 到 81.53%），进一步印证了” 检索增强在预训练表征覆盖不足时收益最大” 的核心论点。

全量数据对比　表 5 与图 5 给出在 CSTNET 全量训练数据下，RAC-ET 与若干代表性基线的对比结果。RAC-ET 在保持 ET-BERT 骨干冻结的前提下，相较全量微调的 ET-BERT 进一步取得 +3.00% 的准确率提升，并在 Macro-F1 上提升 +2.55%。图 6 进一步以平均少样本提升幅度的形式对比三个数据集，可直观看出 RAC-ET 的收益主要来自分布偏移较大的 IoT-23 与 CICIDS2017 场景。

Table 5: CSTNET 全量数据下的基线对比（%）。RF 与 XGBoost 以 ET-BERT 的 `[CLS]` 特征作为输入。

方法	Accuracy	Macro-F1
随机森林（RF）	30.10	27.37
XGBoost	35.55	33.52
ET-BERT（全量微调）	88.75	89.28
RAC-ET (Ours)	**91.75**	**91.83**

与少样本方法的对比　表 6 进一步在 CSTNET 上对比 RAC-ET 与典型少样本学习方法。在类别充分而数据相对充足的应用识别任务中，原型网络与 RAC-ET 表现相近，二者均明显优于直接微调；

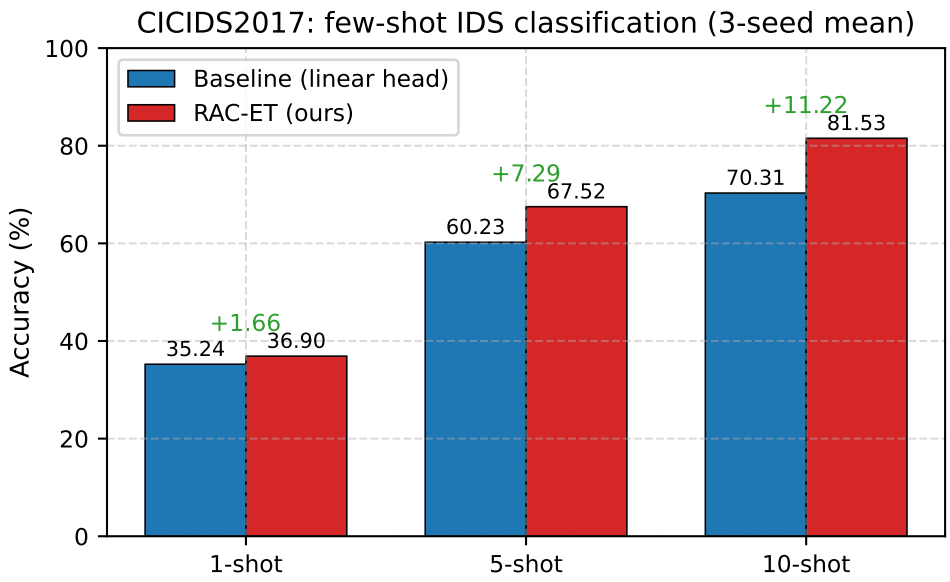

Figure 4: CICIDS2017 数据集上不同 K_s 下 RAC-ET 与基线的准确率对比（3 个种子均值）。使用 78 维 CICFlowMeter 流特征与自监督编码器，RAC-ET 在 5-shot 与 10-shot 下分别取得 +7.29% 与 +11.22% 的显著提升。

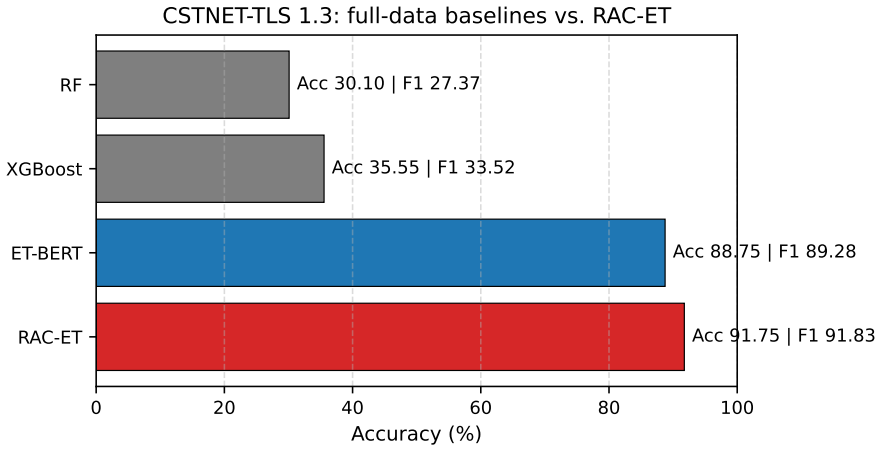

Figure 5: CSTNET-TLS 1.3 全量数据下的基线对比。RAC-ET 在冻结 ET-BERT 骨干的前提下较全量微调基线取得 +3.00% 的准确率增益与 +2.55% 的 Macro-F1 增益。

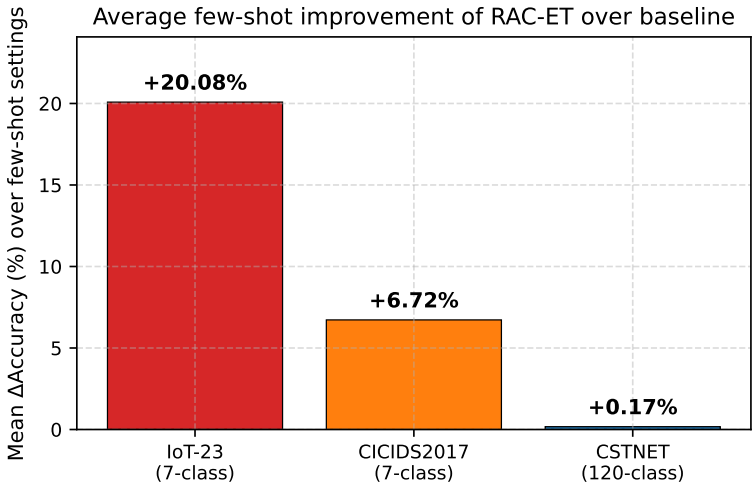

Figure 6: 数据集上 RAC-ET 相对基线的平均少样本提升幅度。IoT-23 与 CICIDS2017（分布偏移较大）上的平均增益显著高于 CSTNET（语义已饱和），印证检索增强主要补偿预训练表征的覆盖盲区。

而在 10-shot 下 RAC-ET 保持竞争力（95.85%），验证了检索增强在该类场景下既不退化也不冗余的特性。

Table 6: CSTNET 上与少样本方法的对比（Accuracy，%）。

方法	1-shot	5-shot	10-shot	20-shot
原型网络（Protonet）	90.61	95.14	96.32	97.82
匹配网络（Matching Net）	90.63	94.65	95.59	96.97
ET-BERT 微调	89.49	95.28	96.73	98.11
RAC-ET (Ours)	90.61	94.85	95.85	97.16

4.3 消融实验

融合组件消融 在 CSTNET 全量数据上，我们依次去除或替换 RAC-ET 融合模块中的关键组件（见表 7 与图 7）。其中”+ Attn+FFN (无残差)” 指保留注意力与 FFN 但去掉残差连接与 LayerNorm。可以看到：

- 简单 MLP 直接拼接查询与检索特征会低于基线（−8.10%），说明朴素的特征串联无法有效过滤检索噪声；
- 单独使用注意力或 FFN 均弱于基线，印证二者需要协同工作；
- 去除残差连接会导致最大幅度退化（−13.50%），表明残差路径对于保持查询特征的主导性与梯度流通至关重要；
- 完整 RAC-ET 达到 91.75%，较 ET-BERT 基线净提升 +3.00%。

Table 7: CSTNET 全量数据上的融合模块消融（%）。

配置	Accuracy	Macro-F1	ΔAcc
ET-BERT 基线	88.75	89.28	–
+ 简单 MLP 融合	80.65	81.02	−8.10
+ 仅注意力	83.40	83.85	−5.35
+ 仅 FFN	83.75	84.21	−5.00
+ Attn+FFN (无残差)	75.25	75.68	−13.50
完整 RAC-ET	**91.75**	**91.83**	**+3.00**

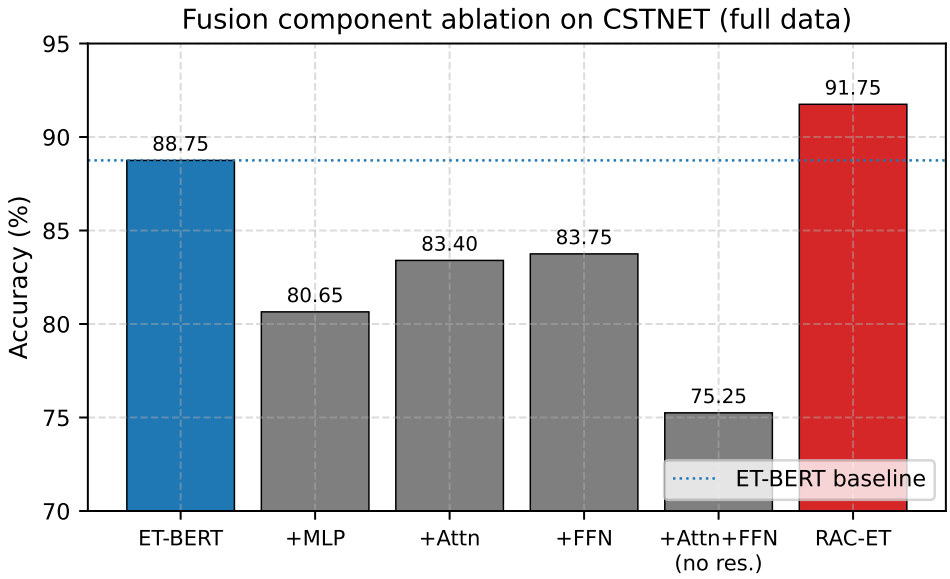

Figure 7: CSTNET 全量数据上各融合组件的消融结果可视化。仅注意力、仅 FFN 或去除残差均低于基线，印证交叉注意力、前馈网络与残差连接必须协同工作，完整 RAC-ET 达到 91.75%。

检索规模 K_r 表 8 与图 8 在 IoT-23 上固定 $K_s = 10$ 分析 K_r 的影响。可以看到 $K_r = 1$ 即取得最高准确率 62.56%，随 K_r 增大，准确率单调下降。这是因为 IoT-23 类间差异较大、类内形态相对集中，Top-1 近邻已可为查询提供较强的判别线索；而继续增加 K_r 会引入更多异类或同类但形态差异较大的样本，反而在融合阶段引入噪声。结合融合机制来看，$K_r = 1$ 的设置在显存占用与推理速度上也最为经济。

Table 8: IoT-23（$K_s = 10$）上 K_r 的消融（Accuracy，%）。

K_r	Accuracy
1	**62.56**
3	56.55
5	52.50
10	47.65

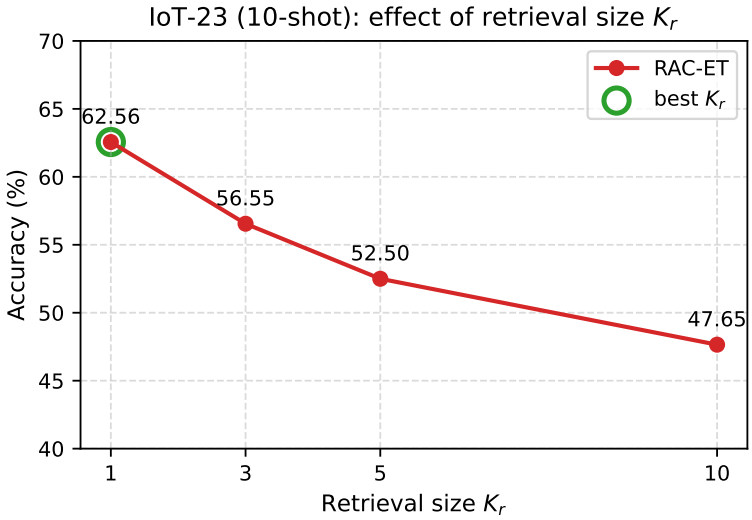

Figure 8: IoT-23（$K_s = 10$）下检索规模 K_r 对准确率的影响。绿色圈出最优配置 $K_r = 1$；继续增大 K_r 会引入异类或形态差异较大的近邻样本，反而在融合阶段引入噪声。

4.4 效率分析

表 9 从可训练参数、训练时间、显存峰值三个维度对比 RAC-ET 与 ET-BERT 全量微调的代价。RAC-ET 冻结 ET-BERT 骨干，仅训练 5.07 M（约 3.69%）参数，训练时间与显存均显著降低；在推理阶段，FAISS `IndexFlatIP` 提供线性级别的精确近邻检索，结合 $K_r = 1$ 的默认设置，整体推理开销相对 ET-BERT 仅小幅增加，可满足常见在线检测的工程约束。

Table 9: 与 ET-BERT 全量微调的效率对比。

指标	ET-BERT 全量微调	RAC-ET (Ours)
总参数量	132.2M	137.3M
可训练参数量	132.2M (100%)	5.07M (3.69%)
CSTNET 训练时间	∼2.5 h	∼12 min
GPU 显存峰值	∼16 GB	∼8 GB

4.5 CICIDS2017 深入分析

为进一步理解 RAC-ET 在 CICIDS2017 上的行为模式，图 9 给出了 10-shot 下各攻击类别的 F1 对比，图 10 展示了 RAC-ET 的混淆矩阵，图 11 则通过 t-SNE 可视化展示了自监督编码器提取的特征分布。

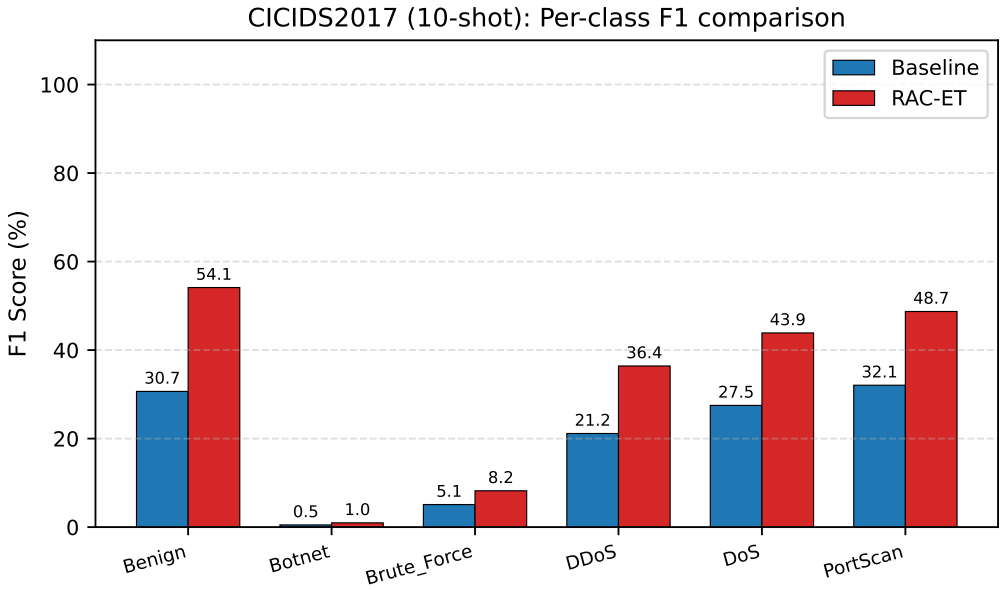

Figure 9: CICIDS2017（10-shot）各攻击类别的 F1 对比。RAC-ET 在 DDoS、PortScan 等攻击类别上的 F1 提升尤为显著，表明检索增强对特征模式明确的攻击类型具有更强的辅助判别效果。

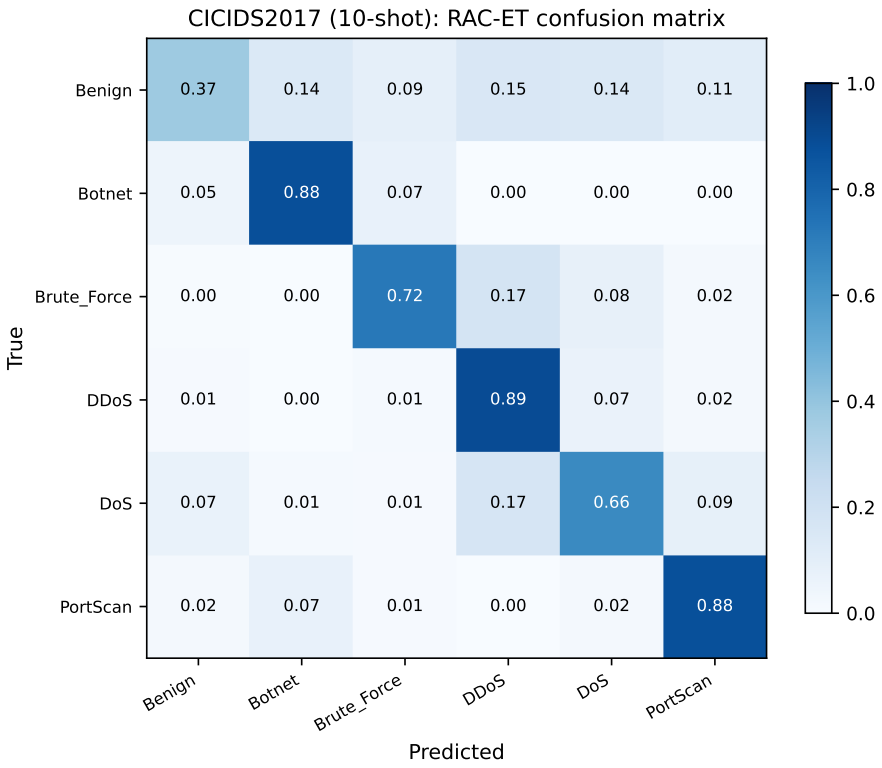

Figure 10: CICIDS2017（10-shot）RAC-ET 的归一化混淆矩阵。对角线值越高表示分类越准确。可以观察到 Benign 与主要攻击类别（DDoS、DoS、PortScan）的区分度较好，少数攻击子类间存在一定混淆。

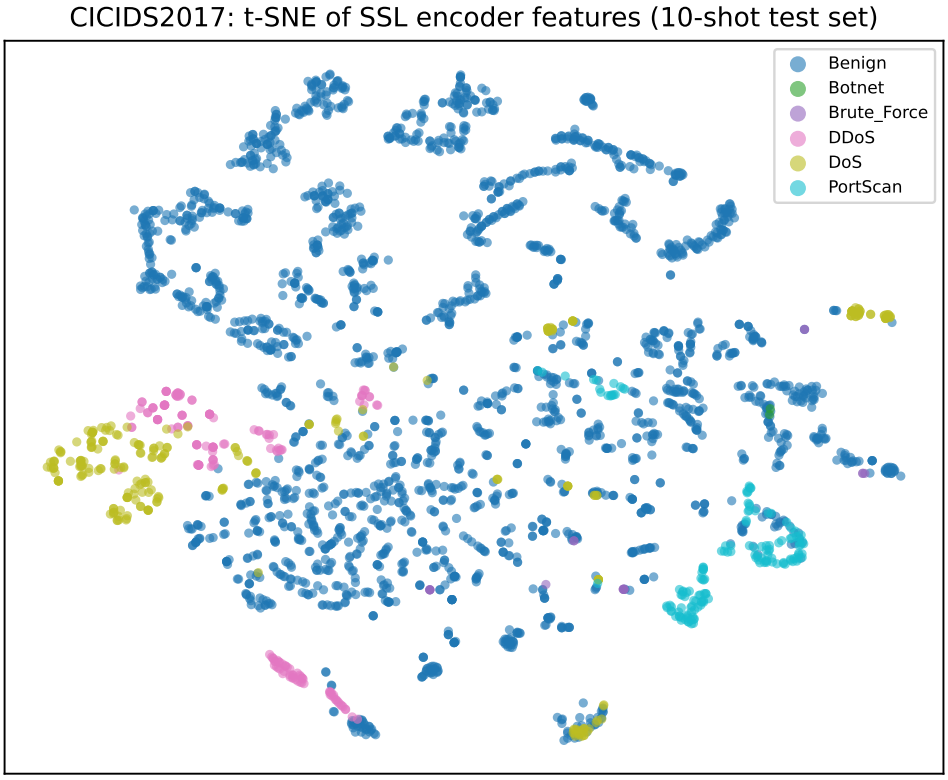

Figure 11: CICIDS2017 自监督编码器特征的 t-SNE 可视化（测试集子样本）。不同颜色代表不同攻击类别。可以观察到自监督预训练已形成初步的类簇结构，但部分类别（如 DoS 与 DDoS）存在重叠区域，这正是 RAC-ET 通过检索近邻提供额外判别线索的作用空间。

4.6 讨论

检索增强何时最有效 综合表 3 与表 8 的结果，可以归纳出 RAC-ET 发挥作用的前提：一是查询样本在语义空间中临近同类样本，使得 Top-K_r 检索具有信息量；二是融合模块能够抑制近邻检索不可避免的噪声。三个数据集的增益梯度可以清晰印证这一论点：CSTNET（ET-BERT 同源预训练）→ CICIDS2017（流特征 + 自监督编码器）→ IoT-23（字节载荷 + 跨域分布偏移），随着预训练表征对目标任务的覆盖程度递减，RAC-ET 的收益递增——检索增强本质上是在补偿预训练表征的覆盖盲区。CICIDS2017 上 10-shot 下 +11.22% 的提升进一步验证了该框架对流统计特征输入同样有效，展现了跨模态泛化能力。

与元学习方法的关系 与原型网络、匹配网络等将支撑样本作为训练阶段的参考不同，RAC-ET 将其建模为推理阶段可动态更新的非参数记忆：一旦出现新类或新变种，只需将新样本写入知识库即可，不必重训融合模块或骨干网络，对实际部署中流量类别演化的友好程度更高。

动态知识库更新与零成本适配 RAC-ET 的架构设计赋予了其一项独特优势：知识库的更新完全独立于模型参数。传统监督式分类器（包括 ET-BERT 微调方案）在面对新型攻击变种或新增类别时，必须收集足够标注样本后重新微调部分或全部参数；而 RAC-ET 只需将新攻击样本通过冻结的编码器提取特征向量，写入 FAISS 索引即可——**无需重训融合模块，无需重训骨干网络**。这意味着在实际部署中，安全运维人员确认一条新型恶意流量样本后，几乎可以实时地将其纳入检测体系，整个过程的计算开销仅为一次前向传播（约 0.01 秒）加一次向量索引更新。这种” 即插即用” 的知识扩展能力，使得 RAC-ET 在攻击模式持续演化的真实网络环境中具有显著的运维效率优势。

局限 首先，CICIDS2017 实验使用流统计特征而非字节载荷，与 CSTNET/IoT-23 的输入模态不完全一致，未来可在获取完整 PCAP 原始数据后统一为字节级输入；其次，当前评估尚未覆盖更严苛的跨域迁移与类增量场景；第三，FAISS 默认使用精确内积搜索，在百万级知识库下仍有优化空间，结合 IVF/HNSW 近似索引的性能—效率权衡值得后续工作专门研究。

5 结论

本文针对加密流量分类中标注样本稀缺、预训练模型少样本泛化能力不足、模型在线更新成本高三个关键问题，提出了一种基于检索增强的轻量级分类框架 RAC-ET。方法侧，本文以冻结的预训练编码器作为特征提取器，构建基于 FAISS 的流量样本知识库，并通过交叉注意力融合查询特征与 Top-K_r 近邻特征，使得分类决策同时利用了参数化语义表征与非参数化样本记忆。实验侧，在 CSTNET-TLS 1.3、IoT-23 与 CICIDS2017 三个基准数据集上系统地验证了框架的有效性：在 CSTNET 全量数据下将准确率从 88.75% 提升至 91.75%（+3.00%），并在 1/5/10-shot 设定下保持 +0.00% ∼ +0.30% 的一致提升；在 IoT-23 少样本设定下平均取得 +20.09% 的显著提升，其中 3-shot 提升达 +28.15%、10-shot 达 +26.23%、20-shot 达 +23.73%；在 CICIDS2017 上使用流统计特征验证了框架的跨模态泛化能力，10-shot 下取得 +11.22% 的提升。效率侧，RAC-ET 仅训练 3.69% 的参数，却取得优于全量微调的准确率，为资源受限环境下的加密流量检测部署提供了一条可行路径。

未来工作将沿三条主线推进：（1）在获取完整原始 PCAP 数据后，统一将 CICIDS2017 的输入模态

从流统计特征升级为字节载荷，并扩展到跨域迁移与类增量场景；（2）探索更高效的近似最近邻索引与可学习的检索器，以在大规模知识库上进一步降低推理时延；（3）研究对抗鲁棒性与隐私保护机制下的检索增强分类，使方法在真实对抗环境中保持稳定。

References

[1] Mohammad Lotfollahi, Ramin Shirali Hossein Zade, Mojtaba Saberian, and Mohammad Javad Siavoshani. Deep packet: A novel approach for encrypted traffic classification using deep learning. *Soft Computing*, 24:1999–2012, 2020.

[2] Shahbaz Rezaei and Xin Liu. Deep learning for encrypted traffic classification: An overview. *IEEE Communications Magazine*, 57(5):76–81, 2019.

[3] Xinjie Lin et al. Et-bert: A contextualized datagram representation with pre-training transformers for encrypted traffic classification. *arXiv preprint arXiv:2202.06335*, 2022.

[4] Imperva Threat Research. 2025 imperva bad bot report. Imperva Report, April 2025. Published April 15, 2025.

[5] Federal Bureau of Investigation. Internet crime report 2024. FBI IC3 Annual Report, April 2025. Published April 23, 2025.

[6] Blake Anderson, Subharthi Paul, David McGrew, and Mustaque Ahamad. Deciphering malware's use of TLS (without decryption). In *Proceedings of the 2018 ACM Internet Measurement Conference (IMC)*, 2018.

[7] Wei Wang, Ming Zhu, Xuewen Zeng, Xiaozhou Ye, and Yiqiang Sheng. Malware traffic classification using convolutional neural network for representation learning. *2017 International Conference on Information Networking (ICOIN)*, 2017.

[8] Gerard Draper-Gil, Arash Habibi Lashkari, Mohammad Saiful Islam Mamun, and Ali A. Ghorbani. Characterization of encrypted and VPN traffic using time-related features. In *Proceedings of the 2nd International Conference on Information Systems Security and Privacy (ICISSP)*, 2016.

[9] Tal Shapira and Yuval Shavitt. Flowpic: Encrypted internet traffic classification is as easy as image recognition. In *IEEE Conference on Computer Communications Workshops (INFOCOM WKSHPS)*, pages 680–687, 2019.

[10] Chang Liu, Li He, Guoliang Xiong, Zhiliang Cao, and Zhen Li. Fs-net: A flow sequence network for encrypted traffic classification. In *IEEE Conference on Computer Communications (INFOCOM)*, pages 1171–1179, 2019.

[11] Xiao Yu, Heng Guo, Yu Qiao, Xiang Wang, Zhen Cao, Qi Wan, Chuan Zhou, Jinxin Zhang, Xianghui Xu, and Jintao Yang. Bert-packet-header: Advanced encrypted traffic classification with contextual information from packet headers. *Applied Intelligence*, 54:124606, 2024.

[12] Siwei Ma, Xiao Wang, Zijian Wang, Xinjie Lin, Jingxian Ren, Lei Wang, Qiben Zhu, Qiang Xu, Kui Hou, Yisong Xue, Yang Yu, Xian Zhang, Hua Lu, Jianyuan Lu, Kuansan Wang, et al. Metarock-etc: Few-shot encrypted traffic classification via fine-grained learning and generalized representation. *Computer Networks*, 251:110581, 2024.

[13] Anjali Sharma, Bhupendra Verma, and M. Tulasi Raman. A survey of encrypted traffic classification in the era of ai and quantum computers. *Computer Networks*, 257:110973, 2025.

[14] Can Ye, Heng Guo, Yichao Ma, Jinxin Zhang, Jiajia Gao, Zhen Cao, Yutong Zhai, Xiang Wang, and Fengtong Xie. Encrypted application traffic classification with auxiliary pretraining and parameter-efficient fine-tuning. *Scientific Reports*, 15:21238, 2025.

[15] Jake Snell, Kevin Swersky, and Richard Zemel. Prototypical networks for few-shot learning. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2017.

[16] Oriol Vinyals, Charles Blundell, Timothy Lillicrap, and Daan Wierstra. Matching networks for one shot learning. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2016.

[17] Chelsea Finn, Pieter Abbeel, and Sergey Levine. Model-agnostic meta-learning for fast adaptation of deep networks. In *Proceedings of the 34th International Conference on Machine Learning (ICML)*, 2017.

[18] Neil Houlsby, Andrei Giurgiu, Stanislaw Jastrzebski, Bruna Morrone, Quentin De Laroussilhe, Andrea Gesmundo, Mona Attariyan, and Sylvain Gelly. Parameter-efficient transfer learning for NLP. In *Proceedings of the 36th International Conference on Machine Learning (ICML)*, 2019.

[19] Edward J. Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Lu Wang, Weizhu Wang, and Weizhu Chen. Lora: Low-rank adaptation of large language models. *arXiv preprint arXiv:2106.09685*, 2021.

[20] Bihan Yang, Hao Song, Huifang Wu, Jianqiang Wei, and Mingzhe Wang. Metamre: Meta-learning and representation enhancement for few-shot encrypted traffic classification. In *Advances in Networked-Based Information Systems, AINA 2023*, pages 476–487, 2023.

[21] Patrick Lewis, Ethan Perez, Aleksandra Piktus, Fabio Petroni, Vladimir Karpukhin, Naman Goyal, Heinrich K”uttler, Mike Lewis, Wen-tau Yih, Tim Rockt”aschel, Sebastian Riedel, and Douwe Kiela. Retrieval-augmented generation for knowledge-intensive NLP tasks. *Advances in Neural Information Processing Systems (NeurIPS)*, 2020.

[22] Vladimir Karpukhin, Barlas Oguz, Sewon Min, Patrick Lewis, Ledell Wu, Sergey Edunov, Danqi Chen, and Wen-tau Yih. Dense passage retrieval for open-domain question answering. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 6769–6781, 2020.

[23] Akari Asai, Zeqiu Wu, Yizhong Wang, Avi Sil, and Hannaneh Hajishirzi. Self-RAG: Learning to retrieve, generate, and critique through self-reflection. In *The Twelfth International Conference on Learning Representations (ICLR)*, 2024.

[24] Urvashi Khandelwal, Omer Levy, Dan Jurafsky, Luke Zettlemoyer, and Mike Lewis. Generalization through memorization: Nearest neighbor language models. *International Conference on Learning Representations (ICLR)*, 2020.

[25] Guoxin Yu, Lemao Liu, Haiyun Jiang, Shuming Shi, and Xiang Ao. Retrieval-augmented few-shot text classification. In *Findings of the Association for Computational Linguistics: EMNLP 2023*, pages 6721–6735, 2023.

[26] Fengnan Li, Elliot D. Hill, Jiang Shu, Jiaxin Gao, and Matthew M. Engelhard. IRIS: Interpretable retrieval-augmented classification for long interspersed document sequences. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 30263–30283, 2025.

[27] Zhenyu Xu et al. A survey on network intrusion detection using deep learning: Current trends, challenges, and future directions. *ACM Computing Surveys*, 56(10):1–38, 2024.

[28] Jeff Johnson, Matthijs Douze, and Herv'e J'egou. Billion-scale similarity search with gpus. *IEEE Transactions on Big Data*, 7(3):535–547, 2019.

[29] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2017.

[30] Iman Sharafaldin, Arash Habibi Lashkari, and Ali A. Ghorbani. Toward generating a new intrusion detection dataset and intrusion traffic characterization. In *Proceedings of the 4th International Conference on Information Systems Security and Privacy (ICISSP)*, pages 108–116, 2018.