基于时序双模特征融合的加密FTP指令细粒度识别方法

付春辉[1]，杨智[1]，郭渊博[2]，李勇飞[1]，金舒原[3]
（1.信息工程大学密码工程学院，河南 郑州 450004；2.海南大学网络空间安全学院，海南 海口 570228；
3.中山大学计算机学院，广东 广州 510275）

摘　要：针对当前网络流量应用层指令细粒度识别方面存在的不足，提出一种基于时序双模特征融合的加密FTP指令细粒度识别方法，并解决IPsec-ESP加密隧道下的FTP指令识别问题。首先，设计基于加密代理的多约束匹配算法，成功实现对ESP加密流量的指令级精确标注；然后，构建包含时序双模特征融合的流量分析框架，从宏观流量模式和微观时序动态两个维度提取特征。实验基于加密代理环境获取真实FTP协议流量数据，实现了对24种FTP细粒度指令（响应）的匹配标注和精确识别，并通过5种机器学习模型的对比实验，验证了该方法的有效性。实验结果表明，该方法在加密FTP指令级分类任务中的准确率达95.4%，显著优于传统单模特征方法，为加密网络环境下的应用层流量识别提供了新的技术路径。

关键词：FTP细粒度指令识别；时序双模特征融合；IPsec-ESP加密隧道；流量分类；机器学习
中图分类号：TP393
文献标志码：A
DOI：10.11959/j.issn.1000–436x.2026075

Fine-grained recognition method for encrypted FTP commands based on fusion of temporal dual-mode features

Fu Chunhui[1], Yang Zhi[1], Guo Yuanbo[2], Li Yongfei[1], Jin Shuyuan[3]
1. School of Cryptography, Information Engineering University, Zhengzhou 450000, China
2. School of Cyberspace Security, Hainan University, Haikou 570228, China
3. School of Computer Science and Engineering, Sun Yat-sen University, Guangzhou 510275, China

Abstract: To address deficiencies in fine-grained identification of application-layer commands in network traffic, an encrypted FTP command recognition method based on temporal dual-modal feature fusion was proposed, solving FTP command identification under IPsec-ESP encrypted tunnels. First, a multi-constraint matching algorithm based on an encrypted proxy was designed to achieve accurate instruction-level annotation of ESP encrypted traffic. Then, a traffic analysis framework with temporal dual-modal feature fusion was constructed to extract features from macroscopic traffic patterns and microscopic temporal dynamics. In experiments, real FTP traffic was obtained in an encrypted proxy environment to realize matching annotation and accurate identification of 24 fine-grained FTP commands (responses). Comparative experiments with five machine learning models verify the effectiveness of the proposed method. The results show that the method achieves 95.4% accuracy in encrypted FTP command-level classification, significantly outperforming traditional single-modal feature methods, providing a new technical approach for application-layer traffic identification in encrypted networks.

Keywords: fine-grained FTP command recognition, temporal dual-mode feature fusion, IPsec-ESP encrypted tunnel, traffic classification, machine learning

收稿日期：2025–12–31；**修回日期：**2026–03–12
通信作者：杨智，zynoah@163.com
基金项目：国家自然科学基金资助项目（No.62472456）
Foundation Items：The National Natural Science Foundation of China (No.62472456)

0　引言

随着信息技术的普及和网络加密技术的快速发展，加密流量已成为保护网络通信安全的重要技术手段。然而，加密技术的广泛应用也为网络流量监控和威胁检测带来了前所未有的挑战。根据Cisco《Annual Internet Report（2018－2023）》，截至2023年约95%的恶意命令与控制流量通过加密通道传输[1]，传统基于深度包检测（Deep Packet Inspection，DPI）的安全防护体系由于无法解析加密内容，导致效能显著退化。攻击者利用加密隧道实施隐蔽通信（如APT攻击）已成为关键基础设施安全的主要威胁[2-3]，亟需基于行为特征的加密流量识别技术。近期中国科学院国家授时中心遭受APT攻击的事件显示，攻击者通过42款特种网攻武器构建多层加密隧道，实现长期隐蔽窃密[4]，这一案例也凸显了加密流量分析的紧迫性和必要性。

文件传输协议（file transfer protocol, FTP）作为企业文件传输、网站维护等场景的核心协议，其明文传输缺陷已通过IPsec-ESP加密技术弥补，但加密环境下的指令级行为难以识别，导致无法检测“未授权文件下载（RETR指令）”“恶意文件上传（STOR指令）”等恶意行为。因此，针对IPsec-ESP加密隧道下FTP细粒度指令识别技术的不足，本文不仅能填补加密流量应用层指令分析的理论短板，更能为关键信息基础设施的隐蔽通信检测提供可落地的技术方案，具有重要的理论价值和实践意义。

当前，加密流量研究主要集中于网络层攻击检测（如DDoS、端口扫描）、协议大类识别（如区分FTP与HTTP）或系统级行为分析（如用户登录、文件访问）[5-6]，缺乏对协议“细粒度指令”（如FTP的USER/RETR）的研究。传统机器学习方法依赖人工特征工程，在面对加密流量动态性时鲁棒性不足，同时面临可解释性不足、对抗样本风险等挑战[7-9]；深度学习模型虽然能自动提取有效的分类特征，但仍存在一定局限性，在跨网络环境下的泛化性不足[10-11]。多模态特征融合技术通过结合不同维度的特征信息，能够提高分类模型的鲁棒性，但在加密环境下的应用研究尚不充分。现有研究显示，基于多视角异构图模型的加密流量分类方法通过块化流量表示和多视角图模型实现对加密流量的高效分类[12]。同时，现有公开数据集普遍存在应用层流量细粒度标签缺失问题[13-14]，难以支撑加密环境下FTP应用层协议的指令级识别任务。

传统协议识别方法仍依赖端口匹配（如FTP协议20/21端口）或明文特征字检测[15-16]，这些方法在加密环境下完全失效，相关研究进一步发展到聚焦行为特征和多模态特征融合，为解决该问题提供了方向，加密流量分析技术发展脉络如图1所示。研究结果表明，通过分析数据包的时间间隔、包长序列等时序特征可以有效识别加密流量的应用类型[17-18]，但针对FTP指令级的细粒度识别尚未形成有效方案，需要优化传统分类模型，并加强多维度特征融合与自动化特征提取。

本文的主要工作如下。

1) 提出一种基于时序双模特征融合的流量识别方法，依据不同时间尺度下的协议行为可分性，融合时间窗口聚合（宏观模式）特征和滑动窗口时序（微观动态）特征，提高模型分类的鲁棒性，解决了加密环境下FTP细粒度指令识别难题。

2) 设计基于加密代理（以IPsec-ESP加密隧道为例）的多约束匹配算法，作为数据集构建过程中获取标签的关键手段，以包序列号连续性约束、时间戳相关性约束和包长差异匹配约束为匹配条件，解决了加密环境下流量精确关联标注的问题。

3) 通过5种机器学习模型的对比实验，实现24种FTP指令（响应）的精确识别，并验证了封装安

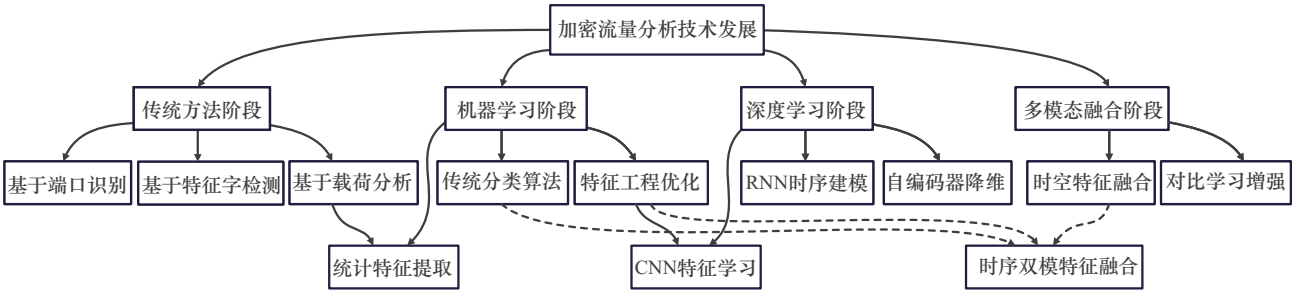


图1　加密流量分析技术发展脉络

全载荷（encapsulating security payload, ESP）加密包指令级标注的准确性，为实时威胁检测提供技术和数据基础，验证了本文方法的有效性。

1 相关工作

1.1 加密流量分类方法研究

加密流量分类模型已从传统的统计方法转向深度学习与多模态融合。早期研究侧重于利用支持向量机（support vector machine，SVM）、随机森林等传统机器学习算法处理流统计特征，虽然具有较好的可解释性，但在处理复杂协议时泛化能力有限。近期研究转向深度学习，通过卷积神经网络（convolutional neural network, CNN）提取空间特征、LSTM/GRU捕捉时序依赖，并进一步演进为基于Transformer的预训练模型和多分支混合架构，显著提升了识别精度。

为兼顾流量的空间与时间特性，Seydali等[19]提出CBS（CNN-Bi-LSTM-SAE）框架，利用1D-CNN提取包内空间特征，Bi-LSTM模型捕捉包间时间序列特征，并结合堆叠自编码器（stacked autoencoder，SAE）提取统计特征，实现流量类型与应用的双重识别。Wang等[20]论证了网络流量本质是一维序列，提出一个完全端到端的1D-CNN分类框架，直接从原始字节流中自动学习特征，省去了人工设计特征步骤。Bu等[21]将NIN（network-in-network）结构引入流量分类，通过并行子网络分别处理包头与包体，利用1×1卷积增强局部非线性表达，显著优于传统CNN。Lin等[11]提出多模态端到端深度学习框架PEAN（packet-level end-to-end attentive network），利用Transformer结构建模包间关系，并引入无监督预训练增强字节表示能力，在处理长序列流量时表现出较强的鲁棒性。

1.2 加密流量特征工程研究

特征工程是加密流量识别的基石，研究重点已从单一特征维度转向协议无关的多模态特征。为应对TLS 1.3协议对服务器名称指示（server name indication, SNI）敏感字段的加密处理，研究者开始屏蔽协议相关字段，转而挖掘协议无关特征（如包到达间隔时间、包大小分布等）、时序动态特征以及高层语义特征。

为应对加密策略的绕过，Akbari等[22]提出一种协议无关的特征工程方法，融合传输层安全协议（transport layer security, TLS）握手包字节、流量时间序列（如包大小、包到达间隔时间等）和流统计特征，并屏蔽SNI敏感字段以增强泛化能力。Chen等[23]针对网络层特征的局限性，引入应用层数据单元（application data unit，ADU）概念，并按请求－响应对切分流量，证明了高层特征在加密环境下的分类优势。Xu等[24]将包长度序列转化为路径签名特征，通过数学变换捕捉流量的几何路径特征，解决高维特征难以训练的问题。然而，上述方法均为会话级特征方法，实现过程需要完整的会话或足够长的包序列进行特征提取，难以支持包级特征的细粒度识别。

1.3 数据集获取与标注技术研究

高质量标注数据的匮乏是加密流量研究的主要瓶颈，针对加密环境下标签缺失和标注成本高等问题，自动化标注与主动学习逐渐成为研究热点。传统数据集（如ISCX系列）存在标签陈旧、场景单一的问题。现有研究通过在用户终端部署插件、利用主动学习减少专家干预以及构建模拟攻击框架，实现大规模、细粒度流量数据的自动化标注。

Kala等[25]提出一种基于终端侧的自动标注方法，通过在用户终端部署浏览器扩展截取加密前的超文本传输协议（hypertext transfer protocol, HTTP）元数据，通过时间与SNI匹配实现超文本传输安全协议（hypertext transfer protocol secure, HTTPS）流量的自动语义关联，解决了加密导致的应用层标签缺失问题。Torres等[26]提出RiskID框架，将主动学习与视觉分析结合，采用不确定性采样策略向用户推荐最不确定的样本进行标注，将专家标注成本降低了一个量级。Poornima[27]同样结合主动学习与监督学习，利用*K*-means聚类辅助自动标注未知pcap文件。Cordero等[28]提出合成攻击注入框架，并提供一个开源工具ID2T，可将合成攻击注入真实网络流量，生成带标签、可复现、无偏差的入侵检测数据集，解决了现有数据集过时、无标签、不可复现的问题。此外，Heng等[29]发布UTMobileNetTraffic2021数据集，通过安卓调试桥（android debug bridge，ADB）遥控手机自动执行应用行为并同步抓包，根据行为脚本执行时间戳与抓包时间窗口对齐，自动打上“应用+行为”标签。该数据集不仅标注了应用类型，还细化了应用行为（如浏览、播放、上传等），为细粒度指令识别研究

提供了标准化基准。这些方法在一定程度上缓解了数据标签获取困难的问题，但在标注准确性和灵活性方面仍有提升空间。

2　相关理论与技术基础

2.1　FTP协议与指令分析

FTP协议作为文件传输领域的经典协议，其控制连接+数据连接的双通道架构是实现文件可靠传输的核心设计[30]。控制连接基于TCP协议在客户端与服务器的21端口建立，全程保持连接状态，负责传输指令（如用户认证、目录操作）与响应码（如230表示认证成功、550表示操作失败），是FTP会话的指挥中枢；数据连接则动态建立，用于传输文件内容、目录列表等二进制或文本数据，分为主动模式（服务器主动向客户端发起连接）和被动模式（客户端向服务器发起连接），两种模式的选择会直接影响流量的端口分布与交互时序特征。

从指令功能维度划分，FTP指令主要分为访问控制（USER/PASS）、目录管理（CWD/PWD）、文件操作（LIST/RETR/STOR）和会话管理（QUIT）4类，不同指令的交互模式（如单向请求、双向响应）、包长分布、时间间隔存在显著差异，为加密环境下的指令识别提供了行为依据，部分FTP控制指令分类及功能分析如表1所示。

1) 访问控制类：包含USER（提交用户名）、PASS（提交密码）指令，是FTP会话的准入环节。此类指令通常以固定格式传输（如“USER admin\r\n”），在明文环境下内容固定，加密后转化为固定长度的ESP数据包，典型长度为60~80B，且两次指令发送的时间间隔稳定（通常为1~3s，取决于客户端交互逻辑），呈现固定包长+稳定时间间隔的双特征标识，成为加密环境下认证阶段识别的关键依据。

2) 目录管理类：涵盖CWD（切换工作目录）、PWD（查看当前目录）、CDUP（返回上级目录）等指令，是FTP会话的空间定位环节。该类指令的交互模式为客户端请求－服务器响应双向通信，例如，CWD指令发送后，服务器会返回250（成功）或550（失败）响应码。由于目录路径长度不固定（如“CWD/docs/2024”与“CWD/data”），指令包长呈现动态变化（范围为50~120B），且请求与响应的时间间隔较短（通常为50~200ms），时序相关性显著，可通过包长序列波动与短时延特征进行区分。

3) 文件操作类：包括LIST（获取目录列表）、RETR（下载文件）、STOR（上传文件）、DELE（删除文件）等指令，是FTP会话的核心业务环节。此类指令触发数据连接建立，形成控制连接指令+数据连接流量的复合流量模式。例如，LIST指令发送后，服务器先通过控制连接返回150（数据连接即将建立）响应码，再通过数据连接传输目录列表数据，表现为控制包+突发数据包包簇的流量特征；RETR/STOR指令则伴随大量数据传输，上行/下行流量占比显著失衡（RETR指令下行流量占比超90%，STOR指令上行流量占比超90%），且数据包包长大多为最大传输单元（maximum transmission unit，MTU）整数倍（典型为1460B），成为加密环境下文件传输行为识别的核心指标。

4) 会话管理类：主要为QUIT（关闭会话）指令，是FTP会话的收尾环节。该指令通常在会话结

表1　部分FTP控制指令分类及功能分析

指令类别	指令名称	功能描述	典型交互模式	加密环境下特征表现
访问控制	USER	提交用户名进行身份验证	客户端→服务器单向请求	固定包长，时间间隔稳定
访问控制	PASS	提交密码完成认证	客户端→服务器单向请求	固定包长，响应时间敏感
目录管理	CWD	切换工作目录	请求－响应交互模式	包长变化，时序相关性高
目录管理	PWD	显示当前工作目录	请求－响应交互模式	固定响应模式
文件操作	LIST	获取文件列表信息	控制连接请求+数据连接传输	双向流量突发特征明显
文件传输	RETR	下载文件操作	控制连接请求+数据连接传输	大数据量传输特征
文件传输	STOR	上传文件操作	控制连接请求+数据连接传输	上行流量主导模式
传输控制	TYPE	设置传输模式	客户端→服务器单向请求	固定模式，频次较低
会话管理	QUIT	关闭连接退出	客户端→服务器单向请求	会话结束标志

束时发送，发送后服务器返回221（会话关闭）响应码，随后控制连接断开。QUIT指令包长固定（约为50B），且是会话内最后一个指令包，其单包发送+后续无流量的时序特征，可作为会话边界识别的关键标志。

在IPsec-ESP加密环境下，FTP指令的明文内容被ESP协议加密，但指令的行为指纹（如包长、时间间隔、流量方向、交互模式等）仍能通过ESP元数据体现，这些特征为加密环境下的FTP指令细粒度识别提供可行性基础，是建立时序双模特征融合的核心前提。

2.2 时序特征分析理论

时序特征分析是通过挖掘时间序列数据中的规律（如趋势、周期性、突发性等），揭示数据背后行为模式的方法，在加密流量识别中占据核心地位。由于加密流量内容不可见，时序特征（如时间间隔、包长序列、流量周期性等）成为揭示流量行为规律的关键手段。FTP指令的交互过程本质上是一个时序事件序列（如USER→230→CWD→250→RETR→150→数据传输→226），具有良好的时序特征。当前主流时序特征分析方法分类对比如表2所示[31]。

传统时序特征提取主要有以下3类技术路线。1) 统计特征分析通过均值、方差、偏度、峰度等量化指标，捕捉数据的集中趋势、离散程度与分布形态，例如，USER指令的包长均值稳定、CWD指令的包长方差较大，此类特征计算简单、可解释性强，是基础分类依据；2) 频域特征分析通过傅里叶变换与小波变换将时域数据转换为频域信息，傅里叶变换适用于识别平稳数据的固定周期（如定期发送的LIST指令），小波变换则擅长捕捉非平稳数据的多尺度特征（如FTP会话各阶段的流量差异），弥补了统计特征无法挖掘周期性规律的不足；3) 时序关联特征分析聚焦数据的动态依赖关系，通过自相关系数识别指令交互的固定模式（如CWD后紧跟PWD的行为），借助转移概率矩阵刻画指令类型或包方向的状态演变（如USER到PASS的高转移概率），精准反映FTP指令的交互逻辑。

深度学习特征分析则通过模型自动学习高阶时序特征，无需人工设计，适用于复杂高维数据。LSTM/GRU模型通过门控机制捕捉长短期依赖关系，适配不同长度的FTP指令序列；Transformer模型基于自注意力机制实现全局依赖关系的并行计算，高效处理长时序会话数据；TCN模型通过一维卷积与扩张机制，兼顾局部特征提取与长序列依赖捕捉，适用于固定长度的包序列分析。此类方法虽然识别精度高，但需大量标注数据训练，可解释性较弱，需根据具体场景选择使用。

从时间尺度的协议行为可分性角度分析，FTP协议的指令交互行为在不同时间尺度下呈现差异化的可分特征。宏观时间尺度（毫秒级及以上）能够反映FTP会话的阶段化行为模式，不同类型指令对应的会话阶段（认证、目录操作、文件传输）在流量统计特征上呈现显著的类间差异；微观时间尺度（微秒级/数据包级）能够刻画单条指令交互的细粒度动态，指令的请求－响应逻辑、包长变化规律、包间时间间隔等特征在该尺度下具备独特的行为指纹。基于此，单一时间尺度的特征提取仅能捕捉FTP协议行为的局部可分信息，而跨时间尺度的双模特征融合能够整合不同尺度下的协议行为可分特征，实现对FTP指令行为的全局刻画，从理论上解决单一尺度特征对协议行为表征不充分的问题。

2.3 多模态特征融合技术

多模态特征融合技术通过结合不同来源和类型的特征信息，提升分类模型的识别性能。加密流量识别本质上可视为网络空间多模态模式识别问题[32]，需融合时序（包间隔）、空间（字节分布）、

表2 时序特征分析方法分类对比

方法类别	技术原理	适用场景	优点
统计特征	计算均值、方差、偏度、峰度等统计量	全局流量特征提取	计算简单，解释性强
自相关分析	计算时间序列的自相关系数	周期性模式检测	有效识别重复模式
频域变换	傅里叶变换、小波变换	频域特征提取	揭示频率分布规律
深度学习	LSTM、Transformer模型	长期动态特征捕捉	自动特征学习，适应性强
异常检测	孤立森林、One-Class SVM	异常流量识别	无监督学习，无需标注

语义（协议状态）等多维特征，以弥补单一模态特征的判别局限性，核心挑战在于处理不同模态数据的异构性（如统计特征与时序特征的量纲差异）与互补性（如宏观流量规律与微观指令动态的协同作用）。

根据融合阶段和模型抽象层级，多模态特征融合可分为数据层融合、特征层融合、决策层融合、模型级融合、注意力融合5个层次，如表3所示。

针对加密FTP指令细粒度、高精度识别需求，结合多模态特征的异构性与FTP指令的行为特点，本文采用特征层加权融合策略，将宏观流量统计特征与微观时序动态特征加权组合，通过特征权重学习算法自动优化融合参数，充分发挥两类特征的互补价值。多模态特征融合技术流程如图2所示。

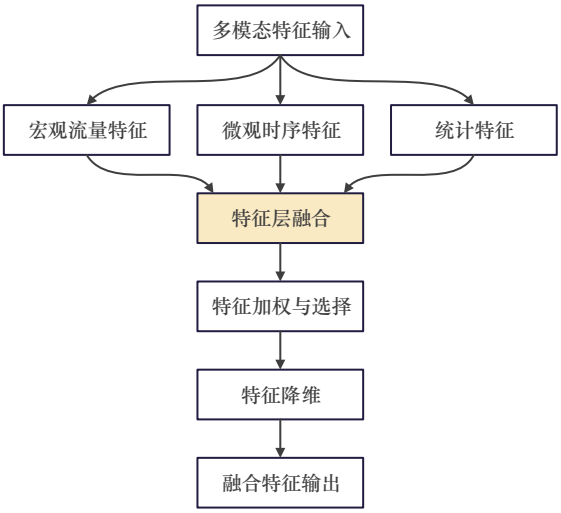


图2　多模态特征融合技术流程

3　时序双模特征融合框架设计

3.1　总体框架设计

本文提出的时序双模特征融合框架采用双流程与模块化设计，分为主处理流程与数据标注流程，包含数据预处理、时序双模特征提取、特征融合、分类识别和多约束匹配标注5个核心模块。主处理流程实现时序双模特征提取与分类识别，充分考虑IPsec-ESP加密环境的特点，从包长序列的时序特征入手，重点解决加密载荷的不可见性带来的流量识别难题。数据标注流程设计了基于加密代理的多约束匹配算法，实现对ESP加密流量的指令级精确标注，为模型的分类训练提供数据标签。时序双模特征融合框架如图3所示。

时序双模特征融合框架采用双路径特征提取策略，宏观流量统计特征提取采用时间窗口聚合方法，通过设定连续固定时间间隔的时间窗口，对流量数据进行多时隙分段统计分析。微观时序动态特征提取采用滑动窗口方法，以连续的数据包序列为分析单元，以固定的窗口大小和步长进行时序特征提取。

其中，数据预处理模块负责流量清洗、包序重组和时间戳对齐，并基于应用层协议分类和加密代理通道分析，将原始流量划分为ESP加密流量和FTP明文流量；时序双模特征提取模块分别从宏观和微观维度捕捉流量特性；特征融合模块采用加权融合策略整合双模特征；分类识别模块进行机器学习模型的训练和分类，实现FTP加密指令的细粒度识别。

框架的创新性体现在3个方面。1) 依据不同时间尺度下的协议行为可分性，设计时序双模特征提取策略，同时捕捉宏观流量模式和微观时序动态；2) 设计基于加密代理的多约束匹配算法，将该算法作为数据集构建的核心技术，解决加密环境下的数据标注难题；3) 通过特征层融合策略实现不同维度特征的互补增强，显著提升分类性能。

3.2　多约束匹配标注

缺少标注数据一直是加密流量识别研究的难题[28,33]，本文设计了基于加密代理的多约束匹配算法，通过解析加密代理前后明文与密文包的特征关

表3　多模态特征融合层次对比

融合层次	融合方式	技术特点	优点	缺点
数据层融合	原始数据直接拼接	早期融合，保留完整信息	信息损失少	数据异构性问题
特征层融合	特征向量加权组合	中期融合，最常用	灵活性高	特征对齐困难
决策层融合	分类结果投票集成	后期融合，模块化	容错性强	信息利用不充分
模型级融合	多模型联合训练	端到端学习	自适应性强	训练复杂度高
注意力融合	注意力机制加权	动态特征选择	重点突出	需要大量数据

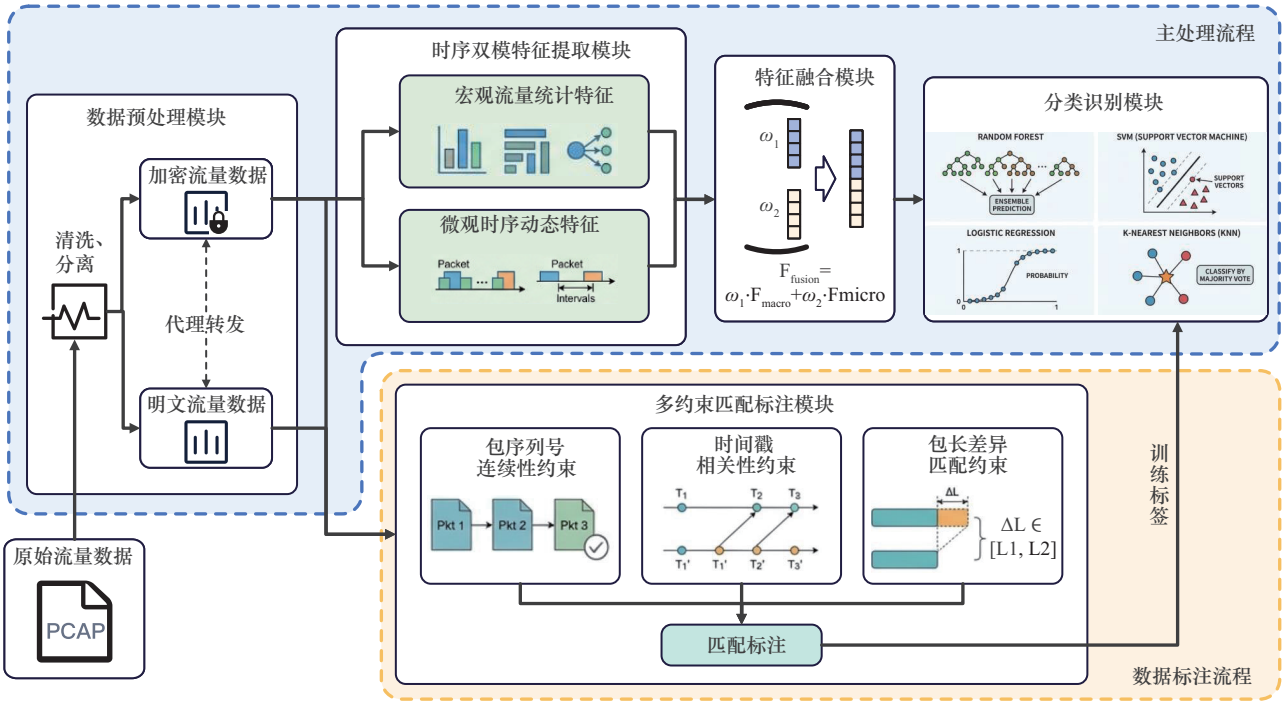


图3 时序双模特征融合框架

系，实现ESP加密流量与原始FTP指令的精确关联标注，解决加密环境下无法直接解析流量内容的数据标注难题，是数据集构建的核心过程。具体而言，通过加密代理环境收集原始明文FTP流量和对应的IPsec-ESP加密流量，并通过多约束匹配算法建立两者之间的精确对应关系。

在加密代理环境中，私有网络与加密代理之间以明文数据进行通信，加密代理与公有网络（服务器）之间以加密数据进行通信，加密代理负责数据的封装与解封装，该环境下的流量数据包结构、包长和时序等关系如图4所示。私网内明文包为标准IP结构（IP头+TCP/UDP头+数据载荷），经ESP封装后，公网传输的密文包结构变为新IP头+ESP头+加密载荷（原始IP包）+ESP尾+认证数据。密文包长在明文基础上增加了ESP头、尾、认证数据及加密填充字节，因此密文包长略大于对应明文包。从时序维度分析，私网出站明文包以$T_{outbound}$时序发送，经ESP代理处理产生时延T_{delay}后，公网密文包时序为$T_{outbound}+T_{delay}$，公网入站密文包以$T_{inbound}$时序到达，经解密解封装后，私网明文包时序为$T_{inbound}+T_{delay}$。

真实网络环境具有一定复杂性，网络波动会造成数据包的乱序和重传，同时ESP加密隧道中会有除FTP协议以外的其他协议流量或控制流量，因此，实际捕获的明密文包不是一一对应关系，两者均存在无法匹配的冗余包。从理论角度分析，该加密代理环境支持任意应用协议的加密数据标签获取，同时支持对公开数据集的加密封装，以获取对

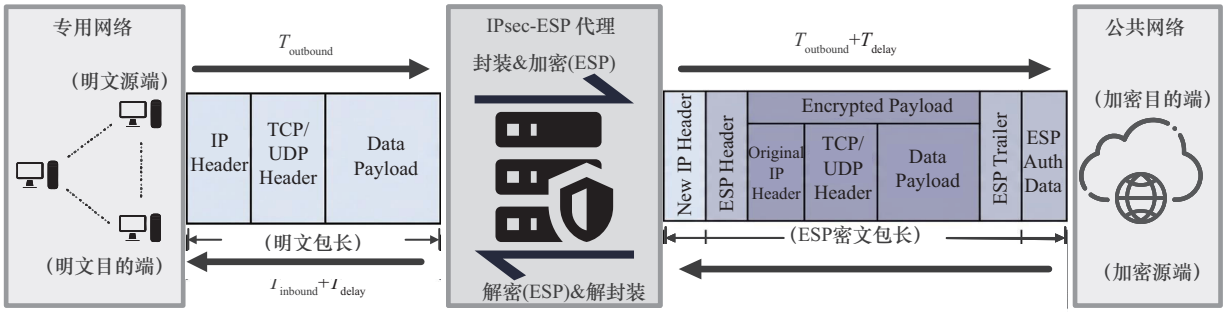


图4 加密代理示意图

应的加密数据集，具有较大的灵活性。

多约束匹配算法基于包序列号连续性约束、时间戳相关性约束和包长差异匹配约束3个核心约束条件，确保标注数据的准确性和可靠性。设明文FTP流量序列为$P=\{p_1,p_2,\cdots,p_m\}$，ESP加密流量序列为$E=\{e_1,e_2,\cdots,e_n\}$，匹配函数$M(p_i,e_j)$表示包p_i与包e_j的匹配程度，其表达式为

$$M(p_i,e_j)=w_1\cdot C_{\text{seq}}(p_i,e_j)+w_2\cdot C_{\text{time}}(p_i,e_j)+w_3\cdot C_{\text{len}}(p_i,e_j) \tag{1}$$

式(1)是加密流量与明文流量精确关联标注的核心计算式，用于量化判断某一ESP加密流量包e_j是否对应某一明文FTP流量包p_i，通过加权融合3个约束条件的匹配度，最终得到两者的整体匹配分数$M(p_i,e_j)$，分数越高，对应关系越可靠。其中，权重系数$w_1+w_2+w_3=1$，取值范围均为$[0,1]$。结合IPsec-ESP加密隧道的流量特征与FTP协议交互特性，包序列号连续性是明密文包匹配的基础，ESP加密对数据包序列号的连续性无影响，其对匹配的贡献度最高。时间戳相关性易受网络微小时延影响，包长差异会由于ESP封装固定头信息产生恒定偏移，两者对匹配的贡献度次之，因此，初始值设置为$w_1=0.4$，$w_2=w_3=0.3$。$C_{\text{seq}},C_{\text{time}},C_{\text{len}}$分别为3个约束条件的匹配度函数，包序号连续性约束的表达式为

$$C_{\text{seq}}(p_i,e_j)=\exp\left(-\frac{\left|\left(\Delta\text{seq}_p\right)_i-\left(\Delta\text{seq}_e\right)_j\right|}{\sigma_{\text{seq}}}\right) \tag{2}$$

时间戳相关性约束的表达式为

$$C_{\text{time}}(p_i,e_j)=\exp\left(-\frac{\left|\left(\Delta\text{time}_p\right)_i-\left(\Delta\text{time}_e\right)_j\right|}{\sigma_{\text{time}}}\right) \tag{3}$$

包长差异匹配约束的表达式为

$$C_{\text{len}}(p_i,e_j)=\exp\left(-\frac{\left|\left(\Delta\text{len}_p\right)_i-\left(\Delta\text{len}_e\right)_j\right|}{\sigma_{\text{len}}}\right) \tag{4}$$

上述匹配函数均基于指数衰减函数设计，核心逻辑是通过量化明文与加密流量特征的差异，计算两者的匹配度。取值范围为0~1，差异越小，则匹配度越接近1。以C_{seq}为例，具体介绍如下。

1) $\left(\Delta\text{seq}_p\right)_i$：明文FTP流量包$p_i$与其前一包的序号差值。

2) $\left(\Delta\text{seq}_e\right)_j$：加密流量包$e_j$与其前一包的序号差值。

3) $\left|\left(\Delta\text{seq}_p\right)_i-\left(\Delta\text{seq}_e\right)_j\right|$：$\left(\Delta\text{seq}_p\right)_i$与$\left(\Delta\text{seq}_e\right)_j$序号差值的绝对误差，误差越小，说明包序列的连续性越一致。

4) σ_{seq}：序号差值的标准差，用于归一化误差，避免量纲影响。

5) 指数函数$\exp(-x)$：将误差转化为0~1的匹配度，当误差趋于0时，匹配度趋于1。当误差增大时，匹配度快速衰减。

C_{time}（时间戳相关性）、C_{len}（包长差异）的逻辑同理，分别从流量包的时间间隔一致性、包长度差异两个维度补充匹配依据，C_{seq}、C_{time}、C_{len}三者共同确保标注的准确性。

匹配判定自适应阈值设为T，用于判定明密文包是否为有效匹配对。首先，对实验环境中已完成人工标注的100组明密文匹配包对计算匹配分数M，得到匹配分数样本集T_{train}；随后，对T_{train}进行核密度估计拟合分布特征，取样本集T_{train}的90%分位数作为阈值初始值T_0，结合实时匹配的流量特征分布进行自适应调整，自适应计算式为

$$T=T_0\times\left(1+\alpha\times\frac{\sigma_M}{\mu_M}\right) \tag{5}$$

其中，μ_M为当前批次待匹配包对匹配分数的均值；σ_M为当前批次待匹配包对匹配分数的标准差；α为调节系数，取值为0.05，用于平衡阈值的稳定性与自适应性。阈值T的取值范围约束为$[0.6,0.95]$，避免由于网络波动导致阈值过度偏离合理区间。当$M(p_i,e_j)>T$时，判定为有效匹配对；否则，判定为无效匹配。多约束匹配算法的伪代码如算法1所示。

算法1　多约束匹配算法

输入　明文FTP流量序列$P=\{p_1,p_2,\cdots,p_m\}$，ESP加密流量序列$E=\{e_1,e_2,\cdots,e_n\}$，自适应阈值T

输出　明文－加密包匹配对(p_i,e_j)序列$\text{SEQ}_{(p,e)}$，未匹配明文包P_{void}，未匹配加密包E_{void}

1) 初始化P_{void}、E_{void}、$\text{SEQ}_{(p,e)}$

2) for i = 1:1:m

//按顺序为每个明文包匹配最佳的加密包

3) Re_{best} = None，记录包p_i方向D_{p_i}

4) for j = 1:1:n*

//n*为当前加密流量序列长度

5) 记录包e_j方向D_{e_j}

6) if D_{p_i} ! = D_{e_j} then

7) continue

8) else

9) 根据式(1)计算$M\left(p_i,e_j\right)$

10) if $M\left(p_i,e_j\right) > T$ then

11) $\text{Re}_{\text{best}} \leftarrow e_j$

12) break

13) end if

14) end if

15) end for

16) if Re_{best} == None then

17) 将p_i添加到P_{void}

18) else

19) 将$\left(p_i,\text{Re}_{\text{best}}\right)$添加到$\text{SEQ}_{(p,e)}$

20) 从E中删除Re_{best}

21) end if

22) 根据式(5)按批次更新自适应阈值T

23) end for

24) $E_{\text{void}} \leftarrow E$

匹配算法的优化策略包括动态权重调整和自适应阈值设定，权重与阈值在匹配过程中实时更新，更新策略如下。1）权重更新：以100个包对为一个更新批次，计算每个约束条件在该批次中有效匹配对的贡献度$S_k\left(k = \text{seq,time,len}\right)$，$S_k = \dfrac{\sum_{M \geq T} C_{\text{k}}}{\sum_{\text{all}} C_{\text{k}}}$，再按$W_k' = \dfrac{S_k}{\sum_{k=1}^{3} S_k}$(6)对权重进行归一化更新，确保更新后仍满足$w_1' + w_2' + w_3' = 1$，使权重能够适配不同网络环境下的特征变化；2）阈值更新：每个批次完成匹配后，按上述阈值自适应调整计算式更新T，同时保留阈值上下限约束，保证匹配判定准确合理。

多约束匹配算法的误匹配率受网络传输特性影响，高丢包率、数据包乱序等复杂环境会破坏明密文包特征关联性，降低匹配准确率，其误匹配率变化趋势与算法抗干扰机制高度相关。高丢包率会破坏包序列号的连续性，导致该约束的匹配贡献度降低，误匹配率随丢包率提升呈线性缓慢增长趋势。在常规网络丢包场景下，算法可通过批次化权重更新提升时间戳、包长特征的匹配权重，弥补序列号特征的失效问题，维持低误匹配率。即使丢包情况加剧，依托三约束的互补特性，误匹配率仅小幅上升，不会出现指数级增长。数据包乱序会造成明密文包时序特征错位，使得时间戳相关性约束的匹配精度下降，误匹配率随乱序程度提升呈小幅增长态势。算法先通过流量方向预过滤剔除方向不一致的包对，且时间戳相关性约束基于指数衰减函数设计，对小幅时序偏差天然具备容忍性，即使网络存在一定程度的乱序，误匹配率仍处于可控范围，无显著波动。此外，由于多约束匹配算法选择的特征与协议无关，多协议混合环境不会对匹配结果产生显著影响，仅增加无效包对的匹配次数。

整体而言，依托三约束互补的匹配机制及动态权重、自适应阈值的抗干扰设计，3类复杂场景下算法的误匹配率均呈可控增长趋势，无突变性上升，可满足加密FTP流量指令级标注的精度要求。同时，本文对匹配结果开展了分层人工抽样验证，基于FTP指令类型分布、未匹配的明文包P_{void}和加密包E_{void}进行抽样，以明文FTP指令交互逻辑为基准对标验证匹配正确性，避免明显的匹配误差。

3.3 时序双模特征提取

3.3.1 宏观流量统计特征提取

宏观流量统计特征提取采用时间窗口聚合方法，通过设定连续的时间窗口对加密流量数据进行分段统计分析，核心目标是捕捉FTP会话级别的全局行为模式，如文件传输阶段与控制命令阶段的流量差异、指令交互的宏观统计规律等，为细粒度指令识别提供全局特征支撑。

时间窗口选择策略采用多时隙分析方法，设置连续固定时间间隔的时间窗口，组合起来能够覆盖短、中、长多个时间窗口尺度。短时间窗口可用于捕捉瞬时流量变化，中时间窗口可用于分析指令交互模式，长时间窗口可用于识别会话级行为特征，

这种多时隙设计确保能够全面捕捉FTP流量的宏观时序特性。

宏观特征提取的关键在于多尺度时间窗口分析和高级统计特征计算，宏观流量特征及计算说明如表4所示。通过分析不同时间粒度下的流量变化规律，能够有效区分FTP会话的不同阶段（认证阶段、命令交互阶段、数据传输阶段）。研究结果表明，宏观流量特征对于会话级别的行为识别更重要，能够有效区分不同的应用类型。

特征标准化处理采用Z-score方法，确保不同量纲的特征具有可比性，表示为

$$X_{\text{norm}} = \frac{X - \mu}{\sigma} \tag{7}$$

其中，μ为特征均值，σ为标准差。这种处理方式消除了特征尺度的差异，为后续特征融合和分类模型训练奠定了基础。

3.3.2　微观时序动态特征提取

微观时序动态特征提取采用滑动窗口技术，以连续的数据包序列为分析单元，重点捕捉FTP指令交互的细粒度时序模式。这种方法能够识别加密环境下仍保持的微观行为特征，如命令－响应的时延模式和包长序列变化规律。

滑动窗口设计采用固定窗口滑动策略，通过合理设置窗口大小与滑动步长，平衡时序信息的完整性与特征提取的实时性和计算效率。窗口大小需保证包含单条FTP指令交互的完整数据包序列，同时预留少量冗余以适配网络波动导致的包重传或乱序现象。滑动步长采用细粒度设计，最大限度保留数据包的时序连续性，避免由于步长过大丢失指令交互的微观动态特征。

微观特征计算主要包括包间时间间隔序列、包长变化序列和流量方向序列3类核心特征。包间时间间隔序列反映指令交互的响应速度特征，包长变化序列体现不同指令的数据传输特性，流量方向序列刻画控制连接与数据连接的交互模式。图5展示了微观特征提取数据流程。

对包长差分结果进行自相关分析，可以量化包长变化的时间关联性。不同应用及其指令的包长变化逻辑可能存在显著差异，其包长差分的自相关特征在分类中具备一定的区分性。

表4　宏观流量特征及计算说明

特征类别	特征名称	计算式	物理意义
流量统计特征	包数量	$N = \sum_{i=1}^{n} I(\text{pkt}_i)$ 其中，$I(\text{pkt}_i)$为指示函数，若存在数据包pkt_i，则取1；否则，取0；n为窗口内数据包的最大可能数量	窗口内总数据包数
流量统计特征	总字节数	$B = \sum_{i=1}^{n} \text{len}(\text{pkt}_i)$	窗口内总流量字节数
流量统计特征	平均包长	$\bar{L} = \frac{1}{n}\sum_{i=1}^{n} \text{len}(\text{pkt}_i)$	平均每个包的大小
流量统计特征	包长方差	$\sigma_L^2 = \frac{1}{n}\sum_{i=1}^{n}\left(\text{len}(\text{pkt}_i) - \bar{L}\right)^2$	包长分布离散程度
流量方向特征	上行下行比	$R = \frac{B_{\text{up}}}{B_{\text{down}}}$ 其中，B_{up}为窗口内上行流量总字节数，B_{down}为窗口内下行流量总字节数	上行下行流量比例关系
流量突发特征	突发性指标	$\text{Burst} = \frac{\max(\text{IAT})}{\min(\text{IAT})}$ 其中，IAT为包间时间间隔序列，$\max(\text{IAT})$为最大时间间隔，$\min(\text{IAT})$为最小时间间隔	流量突发程度度量
时序特性	自相关系数	$\text{ACF}(k) = \frac{\sum_{t=1}^{n-k}\left(X_t - \bar{X}\right)\left(X_{t+k} - \bar{X}\right)}{\sum_{t=1}^{n}\left(X_t - \bar{X}\right)^2}$ 其中，X_t为t时刻的时序特征值，$\bar{X}$为时序特征均值，k为滞后阶数，n为时序长度	时间序列相关性 （用于衡量滞后k步的特征关联程度）

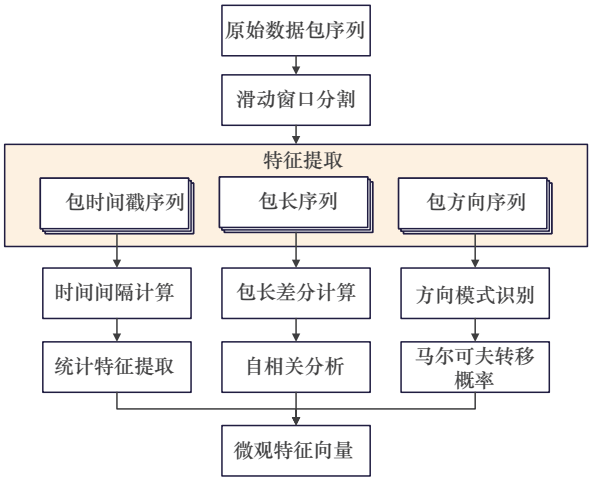


图5　微观特征提取数据流程

方向模式识别的核心目标是从无序的方向序列中，提取重复出现的子序列（模式）或周期性波动特征，这些模式往往与应用层协议逻辑（如请求－响应机制）强相关。包方向序列本质是离散时间序列，且当前方向的取值（如1、0）往往仅与前一个方向相关（满足无后效性），这恰好符合马尔可夫链的核心假设。通过计算马尔可夫转移概率，可量化从一个方向转移到另一个方向的可能性，揭示方向变化的内在规律。

4　实验设计与结果分析

4.1　实验环境与数据集

本文构建了完整的实验环境来验证时序双模特征融合方法在加密FTP指令识别任务中的有效性。实验环境采用高性能计算平台，以满足大规模加密流量数据的处理和分析需求。

实验环境采用虚拟机部署，具体如图6所示，客户机（主机1）、代理服务器（主机2）、FTP服务器（主机3）均为Ubuntu 20.04操作系统，代理服务器与FTP服务器通过StrongSwan5.9采用隧道模式建立IPsec-ESP加密通道，并基于IKEv1协议建立安全关联。其中，IKE和ESP阶段均选用3DES加密算法结合MD5哈希认证，并基于1 024位参数进行DH密钥协商，预存8位PSK。本地主机使用Wireshark 3.6和tcpdump工具对虚拟网卡vnet8进行流量捕获。该环境的加密代理设计使所有客户端（主机1）的数据包均通过代理主机（主机2）与外界或其他主机通信。

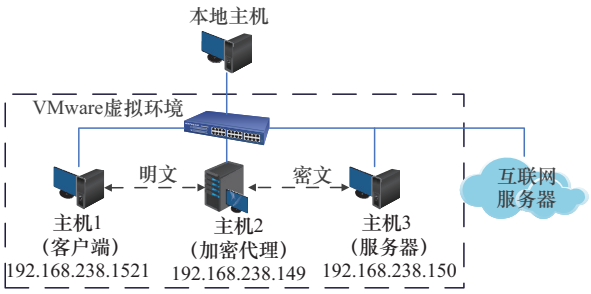


图6　加密代理本地环境部署

自定义数据集构建的关键是通过加密代理环境建立明文FTP指令－ESP密文包的一一对应关系，这也是多约束匹配算法的基础。通过分离实验环境捕获的明密文流量，运用多约束匹配算法，实现高质量的数据标注，确保加密流量与原始FTP指令的精确对应关系。图7展示了加密代理环境下的明文FTP指令包与ESP密文包。

FTP协议指令数据由上述实验环境中客户端（主机1）通过脚本批量生成。实验数据截取时间戳范围为1760152674.118~1760153101.478，持续时间为427.360s。数据集统计信息如表5所示。

该数据集中共有26种指令类型（其中，DATA与other指令非本研究分类对象），完整覆盖FTP协议从“连接建立－指令交互－会话结束”的全链路，可划分为以下3类。

1) FTP控制指令：包含USER、PASS、CWD、PASV、PORT、LIST、NLST、RETR、STOR、DELE、RNFR、REST、QUIT共13种指令，用于实现FTP核心交互（认证、目录操作、文件传输、会话管理），是指令识别任务的核心分析对象。

2) FTP响应码：包含150、200、220、221、

13.335428000	192.168.238.150	21	本地	192.168.238.152	58786	本地	FTP_CTRL		104
13.335524000	192.168.238.152	58786	本地	192.168.238.150	21	本地	TCP		70
13.335611000	192.168.238.149	-	本地	192.168.238.150	-	本地	ESP	ipsec-esp	122
13.437141000	192.168.238.152	58786	本地	192.168.238.150	21	本地	FTP_CTRL		86
13.437781000	192.168.238.149	-	本地	192.168.238.150	-	本地	ESP	ipsec-esp	138
13.438133000	192.168.238.150	-	本地	192.168.238.149	-	本地	ESP	ipsec-esp	122
13.438334000	192.168.238.150	21	本地	192.168.238.152	58786	本地	TCP		70
13.440682000	192.168.238.150	-	本地	192.168.238.149	-	本地	ESP	ipsec-esp	146
13.440790000	192.168.238.150	21	本地	192.168.238.152	58786	本地	FTP_CTRL		93
13.440879000	192.168.238.152	58786	本地	192.168.238.150	21	本地	TCP		70
13.440952000	192.168.238.149	-	本地	192.168.238.150	-	本地	ESP	ipsec-esp	122

图7　加密代理环境下的明文FTP指令包与ESP密文包

表5　　数据集统计信息

指令类型	包数量/个	占比	一级采样数量/个	二级采样数量/个	采样比例
DATA	883 998	—	2 000	100	0.011%
other	7509	—	2000	100	1.332%
200	1034	11.105%	1 034	100	9.671%
TYPE	1034	11.105%	1 034	100	9.671%
150	888	9.537%	888	100	11.261%
227	888	9.537%	888	100	11.261%
PASV	888	9.537%	888	100	11.261%
226	881	9.462%	881	100	11.351%
250	741	7.958%	741	100	13.495%
CWD	741	7.958%	741	100	13.495%
NLST	485	5.209%	485	100	20.619%
350	268	2.878%	268	100	37.313%
REST	268	2.878%	268	100	37.313%
RETR	235	2.524%	235	100	42.553%
550	144	1.547%	144	100	69.444%
LIST	127	1.364%	127	100	78.740%
RNFR	124	1.332%	124	100	80.645%
220	109	1.171%	109	100	91.743%
USER	109	1.171%	109	100	91.743%
230	109	1.171%	109	100	91.743%
QUIT	77	0.827%	77	100	129.870%
221	77	0.827%	77	100	129.870%
STOR	41	0.440%	41	100	243.902%
DELE	19	0.204%	19	100	526.316%
PORT	12	0.129%	12	100	833.333%
500	12	0.129%	12	100	833.333%
合计	9311	—	9311	2400	—

226、227、230、250、350、500、550共11种状态码，是服务器对控制指令做出的状态反馈（如230=认证成功、550=操作失败），与操作类指令形成“请求－响应”配对（如USER→230、LIST→150+226），能够反映FTP交互逻辑的完整性。

3) 数据连接和其他指令：包含DATA数据连接和other其他指令，区分控制流量与数据流量，两者均非本研究聚焦的FTP指令与响应码核心分类对象，其类别定义本身无固定的指令交互特征，单独标注可避免数据连接和其他指令的大流量包对指令识别模型造成干扰。

从统计结果可见，基础控制与响应指令占主导，如TYPE指令－200响应包数量最多，占比11.10%；特殊操作与异常响应占比较低，如PORT指令－500响应包数量最少，仅占比0.13%，整体呈少数指令占比高、多数指令占比低的分布特征，符合FTP协议的实际交互逻辑。

针对各类别的包数量与占比显著不平衡的问题，设计两级采样策略。首先将样本量大的类别的包采样数量限制在2 000个以下，减少对该类样本特征的重复性提取，提高数据预处理与特征提取效率。随后采用分层抽样策略对各样本均匀采样，低

频样本通过重复采样补充，确保训练数据均衡，避免模型偏向高频类别，提升模型泛化能力。

本文实验数据均基于单一加密代理实验环境与标准FTP协议实现获取，实验硬件环境与加密参数配置如前文所述（图6），FTP客户端为Linux原生FTP工具，MTU配置为1 500 B，上述实验条件构成本文结果的核心适用边界，本文方法当前的最优性能均在该环境与参数配置下实现。本文核心是基于FTP协议与客户端、加密参数无关的原生行为指纹特征（包长分布、包间时间间隔、流量方向、交互模式等），未依赖特定实现的专属特征，因此从理论层面具备应用层指令识别的通用性，但针对多客户端和不同MTU配置、IPsec-ESP加密参数的复杂网络场景，实际应用效果仍有待进一步验证。

4.2 特征提取与模型训练

特征提取阶段采用时序双模特征设计，分别从宏观流量模式和微观时序动态两个维度提取流量序列特征。宏观特征通过时间窗口聚合方法计算，微观特征采用滑动窗口技术捕捉细粒度时序模式。

宏观特征提取设置固定时隙0.000 5 s，并划分10个时隙作为当前包的特征提取窗口。该时隙设置依据实验环境中IPsec-ESP加密代理的传输特性与FTP指令交互的时间粒度确定。本文实验构建的ESP加密代理环境中，数据包转发时延稳定在0.000 1~0.0003 s，结合FTP控制指令单次交互的数据包传输耗时（均值0.000 4 s），0.000 5 s的时隙设置既可满足单时隙对指令数据包的区分需求，又可实现多时隙对完整FTP指令交互数据包的覆盖，同时避免由于窗口过大引入无关流量特征，确保宏观特征对单指令行为的精准表征。在每个时间窗口内计算包数量、总字节数、平均包长、包长方差、上行流量与下行流量比例、流量突发性指标和自相关系数等统计特征。

微观特征提取采用固定大小滑动窗口策略，并基于tsfresh库自动提取时序特征，最优窗口大小设为16个数据包，滑动步长为1个数据包。参数设置结合FTP协议指令交互的数据包序列规律与实验数据集特征确定，实验中加密FTP单条控制指令（含请求－响应）的相关数据包数量通常为4~8个，数据连接相关指令需要额外建立数据连接并传递数据，数据包序列长度也相应增加，选取16个数据包作为窗口大小，可覆盖各类FTP指令的完整交互包序列，同时冗余的少量数据包可包含数据连接的主要部分，并适配网络波动导致的包重传、乱序问题；滑动步长设为一个数据包，旨在最大限度保留数据包的时序连续性，避免由于步长过大丢失指令交互过程中的微观动态特征（如包间时间间隔、包长变化的细粒度规律），该设置经实验验证，保证了特征提取的完整性。微观特征主要包括包间时间间隔序列的统计特征（均值、标准差、极值）、包长变化序列的自相关性和差分特征、流量方向序列的马尔可夫转移概率、时序序列的熵值和复杂度指标等23个特征。

此外，为缓解双路特征提取阶段由于数据量大、特征计算密集导致的效率瓶颈，采用多线程处理模式提升效率，以增强流量识别实时性。首先，根据采样包序列号提取包含时间聚合窗口和包滑动窗口的包序列，将原始流量特征转化为若干个特征序列分片，确保每个分片包含完整的指令交互片段，避免序列缺失或乱序导致特征失真；然后，通过Python ProcessPoolExecutor构建线程池，让每个线程单次只处理一个样本分片的宏观+微观特征提取。

特征融合模块采用特征层加权融合策略，基于特征重要性实现动态权重分配，将两类特征向量整合为统一特征矩阵，从而最大化保留互补信息。通过随机森林模型进行特征重要性评估，计算单类特征中各维度的重要性得分，再通过5折交叉验证测试不同权重组合（宏观权重为0.3~0.7，微观权重为0.7~0.3）对模型性能的影响，最终确定宏观特征权重0.6、微观特征权重0.4为最优配置，该配置既体现宏观特征对会话级行为的主导作用，又保留微观特征对指令细粒度动态的补充价值。在权重分配后，通过公式 $F_{\text{fusion}} = \omega_1 \cdot F_{\text{macro}} + \omega_2 \cdot F_{\text{micro}}$（$\omega_1 = 0.6$为宏观权重，$\omega_2 = 0.4$为微观权重，$F_{\text{macro}}$为宏观特征向量，$F_{\text{micro}}$为微观特征向量）实现特征融合，形成最终的融合特征矩阵。

本文采用SelectKBest对融合后的高维特征（宏观为7×10维＋微观为23维）进行降维，ANOVA F-value作为评分函数，计算式为 $F = \frac{\text{组间均方误差}}{\text{组内均方误差}}$，其中，$F$值越大表示该特征对不同类别的区分能力越强。最终根据F值保留30维特征，有效降低了模型计算复杂度（训练时间减少约

70%）。

分类识别模块将融合后的特征矩阵输入机器学习模型，完成FTP指令分类，同时通过特征重要性分析反向验证特征融合效果。该模块采用Pipeline架构，设计5种机器学习模型的对比实验，具体如下。

1) 随机森林：n_estimators=100，random_state=42。

2) 梯度提升：默认参数，random_state=42。

3) SVM：kernel=‘rbf’，random_state=42。

4) 逻辑回归：max_iter=1000，random_state=42。

5) K近邻：默认参数。

特征标准化处理采用Z-score归一化方法，消除不同特征量纲的影响，确保特征融合时各维度的可比性，为后续分类模型训练奠定了基础。

模型训练采用分层k折交叉验证与正则化策略防止过拟合现象。训练过程中采用5折分层交叉验证监控模型泛化性能，对随机森林、梯度提升等树模型设置最大深度限制，对逻辑回归、SVM模型引入L2正则化项控制模型复杂度，避免模型对训练集过度拟合。最终5种模型均完成有效训练且泛化性能稳定，为性能评估提供可靠的模型基础。

4.3　性能评估与对比分析

性能评估采用多指标综合评估体系，包括准确率、精确率、召回率、$F1$分数和AUC值，以全面衡量各分类模型在FTP指令识别任务中的性能表现。5种机器学习模型性能对比如表6所示。

实验结果表明，基于时序双模特征融合的方法在FTP指令级分类任务中取得了显著效果。其中，梯度提升树（Gradient Boosting）表现最优，准确率达95.8%，AUC值达0.999；随机森林（Random Forest）算法性能紧随其后，两者均显著优于其他对比算法。

表6　5种机器学习模型性能对比（20%测试集）

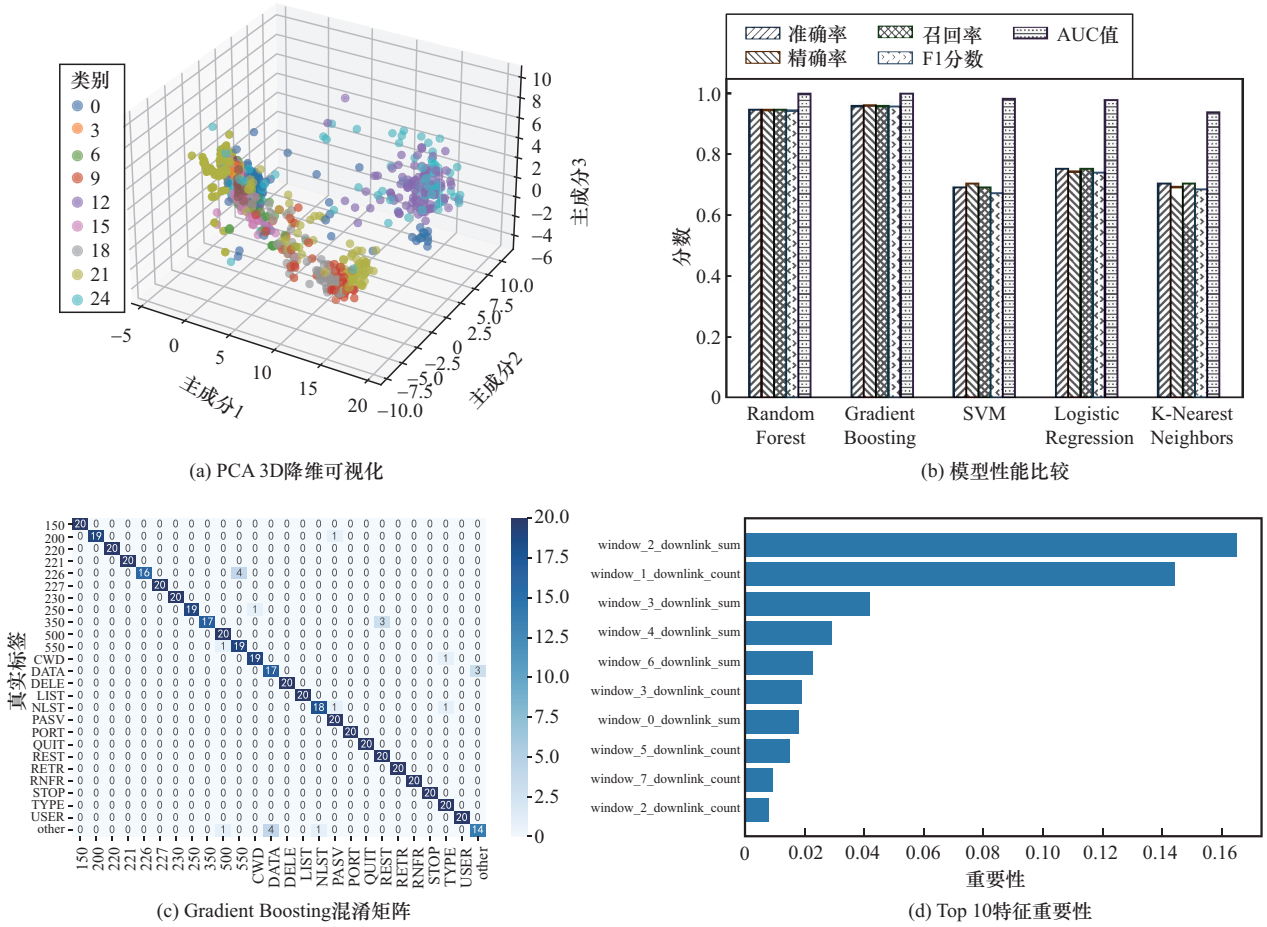
模型算法	准确率	精确率	召回率	$F1$分数	AUC值
Random Forest	94.62%	94.48%	94.62%	94.30%	0.9981
Gradient Boosting	95.77%	95.97%	95.77%	95.72%	0.9991
SVM	69.04%	70.31%	69.04%	67.22%	0.9816
Logistic Regression	75.19%	74.27%	75.19%	73.89%	0.9783
K-Nearest Neighbors	70.38%	69.17%	70.38%	68.49%	0.9373

Gradient Boosting和Random Forest作为集成学习方法，能够处理特征之间的复杂非线性关系，适用于处理高维特征数据，整体分类效果较好；Logistic Regression作为线性模型，处理非线性关系的性能受限，部分类别分类效果较差；KNN依赖距离度量，在高维空间中容易受到“维度灾难”的影响；SVM虽然AUC值较高(0.982)，但准确率较低，原因可能在于核函数选择或参数调优不理想。

通过主成分分析（principal component analysis, PCA）将高维特征投影到三维空间，如图8(a)所示，可以可视化特征分布与类别分离情况，辅助理解数据的可分离性。

图8(c)所示的混淆矩阵揭示了不同FTP指令类别的误分类情况，TYPE、USER、STOR等20种指令（响应）的识别准确率超过90%，仅少数指令出现较低概率的系统性误分类。其中，响应226以20%的概率被误识别为响应550，响应250以10%的概率被误识别为CWD指令，响应350以15%的概率被误识别为REST指令。从特征分布差异角度分析，此类误分类的核心原因是相关指令（响应）的时序双模特征存在显著重叠，226与550均为文件操作类指令的终端响应码，两者宏观特征中的下行流量字节数、包长均值高度重合（226包长均值为72±3 B，550包长均值为70±4 B），微观特征的包间时间间隔均集中在60~80 ms；250是CWD指令的成功响应码，两者滑动窗口包长序列、流量方向马尔可夫转移概率高度相似，且宏观时间窗口突发性指标均小于1.5；350是REST的断点续传响应码，两者微观特征的包长差分自相关系数均在0.85以上，宏观特征的上行下行比均趋近于1。此外，DATA和other类别的识别率相对较低，且存在相互识别错误，原因是两者本身的类别定义模糊，包含多种不同类型的流量，其宏观统计特征离散度大、微观时序特征无固定模式，且并非本实验细粒度指令识别的核心研究对象。

特征重要性分析如图8(d)所示，实验结果表明，宏观流量特征具有主导作用，验证了特征融合动态权重分配的合理性。下行流量特征更重要，反映协议行为模式中下载行为比上传行为更具区分度。近期窗口比远期窗口的重要性高，表明指令（响应）与近期行为模式相关性更强，其中，window2 downlink sum和window1 downlink count具有



(a) PCA 3D降维可视化

(b) 模型性能比较

(c) Gradient Boosting混淆矩阵

(d) Top 10特征重要性

图8 模型分类训练效果

显著的区分能力。

计算效率分析表明，时间窗口特征与包滑动窗口特征提取的时间复杂度均为$O(N)$，模型训练与推理的时间复杂度取决于所选择的机器学习模型，以Gradient Boosting为例，该模型训练的时间复杂度为$O(N)$，推理的时间复杂度为$O(1)$，具备高效推理的基础。为进一步量化验证方法的计算效率与实时处理能力，本文实验基于AMD Ryzen 7 5800H CPU、32GB DDR4内存的硬件环境，采用4线程并行处理策略，对特征提取、模型推理两个阶段的计算效率进行测试。实验结果表明，时间窗口特征提取与滑动窗口特征提取的速度分别为32.83样本/s和187.44样本/s，模型单样本推理速度达52样本/s。结合FTP指令实际交互特性，单条FTP指令的交互间隔通常在50 ms以上，本文方法能够实现对FTP指令交互流量的实时处理。同时，该模型所需的滑动窗口包数量较少、时间窗口时隙较短，进一步保证处理的实时性。

本文设置两组对比实验，分别为Akbari等[22]与Xu等[24]提出的特征工程方法。为保证对比实验的公平性，对相应方法进行针对性适配调整，限定输入信息仅为ESP加密流量元数据（包长、时间戳、流量方向），并满足本文对包级标签、短序列特征的约束条件，具体适配细节如下：1）文献[22]方法需分析包含TCP握手包的完整会话流（包数量≥100），并融合TLS握手字节、SNI字段等信息，本文移除所有与明文、会话级强相关的特征，仅保留其协议无关的时序统计特征（包长分布、包时间间隔、流上行下行比）；将原长会话滑动窗口（窗口大小=100）替换为与本文一致的短窗口（窗口大小=16），基于ESP加密元数据重新计算特征，其余特征计算逻辑保持与原文献一致。2）文献[24]的路径签名特征方法需基于长度不小于200的包序列进行路径签名变换，本文将包序列长度适配为本文微观特征提取的短序列长度（16个数据包），仅使用ESP加密元数据的包长

序列进行路径签名特征计算，相关数学变换与特征降维逻辑保持与文献[24]一致。所有对比方法均使用与本文方法相同的数据集、特征标准化方法（Z-score）及模型训练参数，仅在特征提取环节保留对应文献方法的核心逻辑，同时满足本文的输入与约束条件，确保对比结果的有效性与公平性。对比实验结果如表7所示，不同模型的准确率与*F*1分数对比如图9所示。

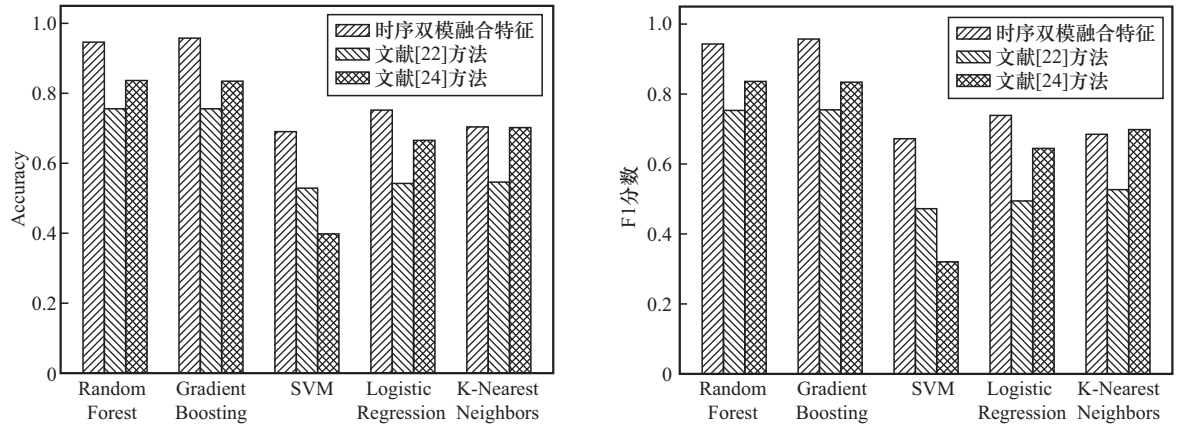
对各组最优模型分类结果进行分析，本文方法的时序双模融合特征明显优于对比方法，准确率达95.77%，*F*1分数达95.72%。文献[22,24]方法在本文特征提取及分类任务中表现欠佳，准确率分别为75.58%和83.65%，*F*1分数分别为75.31%和83.61%。文献[22,24]方法的分类效果分别较对应文献显著降低，主要原因在于文献[22,24]方法为会话级特征方法，要求具备必要的包头信息、交互过程或较长的包序列，如文献[22]需分析包含TCP握手包在内的完整会话流且包数量大于100，文献[24]进行路径特征提取需要的包序列长度为200左右。而本文以加密环境下的应用层协议指令细粒度分类为研究内容，所提供的数据标签为包级标签，适合包级或短序列特征分类，无法支持长序列分类识别。同时，ESP隧道模式加密导致部分包头或控制信息无法解析，因此，文献[22,24]方法应用于本文分类任务的效果一般，证明了本文方法在细粒度识别任务中的有效性和实时性。

4.4　消融实验

消融实验通过逐步移除特征组件和分析参数敏感性，深入探究时序双模特征融合方法中各组成部分的贡献度和参数配置的影响规律。设置两组消融实验，分别使用特征融合前的宏观流量统计特征与微观时序动态特征，以验证特征融合方法的有效性。消融实验结果如表8所示，不同模型的准确率与*F*1分数对比如图10所示。

表7　　**对比实验结果（20%测试集）**

序号	方法	最佳模型	准确率	*F*1分数	AUC值	交叉验证准确率
1	时序双模融合特征	Gradient Boosting	95.77%	95.72%	0.9991	95.58 (±0.52)%
2	文献[22]方法（2022年）	Random Forest	75.58%	75.31%	0.9696	77.42 (±1.04)%
3	文献[24]方法（2025年）	Random Forest	83.65%	83.61%	0.9875	83.62 (±0.38)%

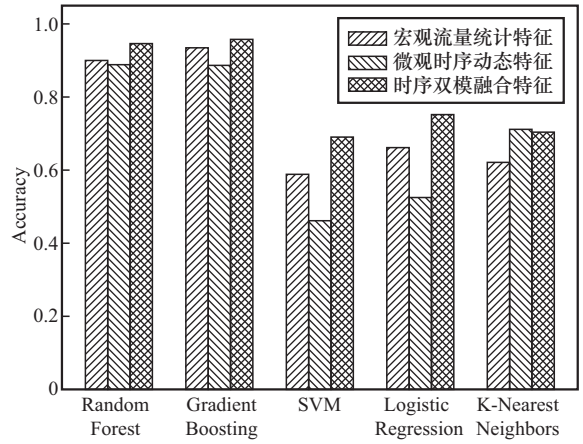


(a) Accuracy对比模型　　(b) F1分数对比模型

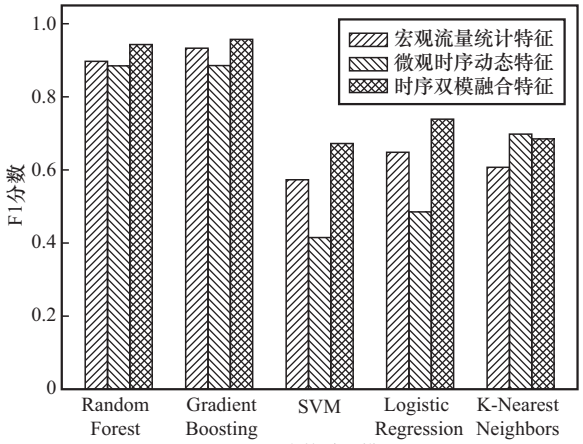
图9　对比实验结果

表8　　**消融实验结果（20%测试集）**

序号	方法	最佳模型	准确率	*F*1分数	AUC值	交叉验证准确率
1	时序双模融合特征	Gradient Boosting	95.77%	95.72%	0.9991	95.58% (±0.52%)
2	宏观流量统计特征	Gradient Boosting	93.46%	93.31%	0.9976	91.85% (±1.02%)
3	微观时序动态特征	Random Forest	88.85%	88.47%	0.9938	89.42% (±0.48%)



(a) Accuracy对比模型



(b) F1分数对比模型

图10 消融实验结果

分析各组最优模型分类结果，时序双模融合特征集表现最佳。时间窗口特征集单独使用准确率为93.46%，*F*1分数为93.31%。包滑动窗口特征集单独使用准确率为88.85%，*F*1分数为88.47%。实验结果表明，基线对照组中的宏观特征和微观特征各自都具有相当的判别能力，双模特征加权融合进一步使准确率提高了3~10个百分点，说明宏观和微观特征包含互补信息，验证了双模特征融合策略的有效性。

从不同FTP指令类别识别的特征贡献来看，宏观流量统计特征对会话级、大流量相关指令的识别起主导作用，如文件传输类RETR/STOR指令、数据连接相关的150/226响应码，依托上行下行比、流量突发性、包长统计等宏观特征，这些指令的单独识别准确率可达92%以上，能够精准捕捉大流量交互的全局行为模式。微观时序动态特征则对细粒度、短序列交互指令的识别表现更优，如目录管理类CWD/REST指令、认证类USER/PASS指令，借助包间时间间隔、包长差分自相关、流量方向马尔可夫转移概率等微观特征，可有效区分指令交互的细粒度时序差异，单独识别准确率达89%以上。上述特征对不同FTP指令类别的差异化贡献，与FTP协议原生的指令交互逻辑高度契合。文件传输类RETR/STOR、数据连接响应150/226等会话级大流量指令，其协议交互本质为海量数据传输，呈现上行下行比失衡、流量突发性显著等宏观流量特征，宏观统计特征可精准捕捉其全局行为模式。而目录管理类CWD/REST、认证类USER/PASS等细粒度短序列指令，属于FTP协议的轻量交互环节，指令传输呈现包长动态变化、请求－响应时序关联紧密等特点，微观时序特征可有效刻画此类细粒度的时序动态差异。宏观与微观特征的贡献差异，本质上是FTP不同功能指令的行为特征与特征提取维度的精准匹配，进一步验证了时序双模特征融合策略对FTP协议指令体系的适配性。

5 结束语

本文针对IPsec-ESP加密隧道环境下的FTP细粒度指令识别问题，系统地提出并验证了基于时序双模特征融合的方法。通过构建包含宏观流量模式和微观时序动态的双模特征框架，结合多约束匹配算法，成功实现加密环境下FTP指令的精确识别，为加密网络环境下的应用层流量分析提供了新的技术途径。

实验结果表明，基于时序双模特征融合的方法在FTP指令级分类任务中取得了显著性能提升。Gradient Boosting算法表现最优，准确率达95.77%，AUC值达0.999，显著优于传统单模特征方法。消融实验验证了特征融合策略的有效性，完整模型准确率较仅用宏观特征提升2.3个百分点，较仅用微观时序特征提升6.9个百分点。同时，研究基于标准FTP协议构建的高质量标注数据集，也为后续跨场景FTP指令识别研究提供了重要支撑。

特别地，本文将研究成果与实际问题紧密结合，在国家关键信息基础设施安全防护场景中展现了重要的应用价值。通过实时识别加密FTP指令，能够有效检测隐蔽的恶意通信行为，为APT攻击检测和威胁情报分析提供技术支撑。该研究成果推动加密流量分析技术的发展，为网络安全监控提供有效的技术手段。

未来研究可在本文基础上进一步拓展与深化。一方面，可将本文方法扩展到TLS1.3、DTLS、WireGuard等更多主流加密协议以及SFTP、FTPS等加密文件传输场景，提升模型的通用性与适用范围；另一方面，结合轻量化神经网络与边缘计算架构，优化特征提取与推理效率，以满足高带宽、低时延网络环境下的实时检测需求。同时，针对对抗样本、流量混淆等规避手段，可引入跨域自适应与自监督学习增强模型鲁棒性，并从单指令识别延伸至多指令序列行为建模，实现更细粒度的威胁意图判定。此外，构建覆盖多协议、多设备、多攻击类型的标准化细粒度加密流量数据集，也将为后续相关研究提供更可靠的数据支撑与统一评测基准。

参考文献：

[1] Cisco. Cisco annual Internet report (2018-2023) [R]. (2024-12-08) [2025-12-31].

[2] Sharma A, Gupta B B, Singh A K, et al. Orchestration of APT malware evasive manoeuvers employed for eluding anti-virus and sandbox defense[J]. Computers & Security, 2022, 115: 102627.

[3] Talabani H S, Abdul Z K, Mohammed Saleh H M. DNS over HTTPS tunneling detection system based on selected features via ant colony optimization[J]. Future Internet, 2025, 17(5): 211.

[4] 中国国家互联网应急中心. 关于国家授时中心遭受美国国家安全局网络攻击事件的技术分析报告[R]. (2025-10-19)[2025-12-31].
National Computer Network Emergency Response Technical Team/Coordination Center of China (CNCERT/CC). Technical analysis report on the cyber attack against the National Time Service Center (NTSC) by the National Security Agency (NSA) [R]. (2025-10-19) [2025-12-31].

[5] Papadogiannaki E, Ioannidis S. A survey on encrypted network traffic analysis applications, techniques, and countermeasures[J]. ACM Computing Surveys, 2022, 54(6): 1-35.

[6] Wang P, Chen X J, Ye F, et al. A survey of techniques for mobile service encrypted traffic classification using deep learning[J]. IEEE Access, 2019, 7: 54024-54033.

[7] Catal C, Giray G, Tekinerdogan B. Applications of deep learning for mobile malware detection: a systematic literature review[J]. Neural Computing and Applications, 2022, 34(2): 1007-1032.

[8] Shen M, Ye K, Liu X T, et al. Machine learning-powered encrypted network traffic analysis: a comprehensive survey[J]. IEEE Communications Surveys & Tutorials, 2023, 25(1): 791-824. [LinkOut]

[9] 付钰, 刘涛涛, 王坤, 等. 基于机器学习的加密流量分类研究综述[J]. 通信学报, 2025, 46(1): 167-191.
Fu Y, Liu T T, Wang K, et al. Survey of research on encrypted traffic classification based on machine learning[J]. Journal on Communications, 2025, 46(1): 167-191.

[10] Zuo M, Guo C Y, Xu H Y, et al. METC: a hybrid deep learning framework for cross-network encrypted DNS over HTTPS traffic detection and tunnel identification[J]. Information Fusion, 2025, 121: 103125.

[11] Lin P, Ye K J, Hu Y S, et al. A novel multimodal deep learning framework for encrypted traffic classification[J]. IEEE/ACM Transactions on Networking, 2023, 31(3): 1369-1384.

[12] Yin J N, Cui L, Hao Z Y, et al. BTRFormer: hierarchical learning of encrypted traffic using a masked autoencoder with block-based traffic representation[C]//Proceedings of the 2025 IEEE 33rd International Conference on Network Protocols (ICNP). Piscataway: IEEE Press, 2025: 1-12.

[13] Najm I A, Saeed A H, Ahmad B A, et al. Enhanced network traffic classification with machine learning algorithms[C]//Proceedings of the Cognitive Models and Artificial Intelligence Conference. New York: ACM, 2024: 322-327.

[14] Bagui S S, Mink D, Bagui S C, et al. Introducing UWF-ZeekData22: a comprehensive network traffic dataset based on the MITRE ATT&CK framework[J]. Data, 2023, 8(1): 18.

[15] Wang Y P, Yun X C, Zhang Y Z, et al. Rethinking robust and accurate application protocol identification[J]. Computer Networks, 2017, 129: 64-78.

[16] Luo J, Chen Z C, Chen W X, et al. A study on the application of the T5 large language model in encrypted traffic classification[J]. Peer-to-Peer Networking and Applications, 2025, 18: 15.

[17] Wang Z H, Yang Y, Wang Y J. A survey of encrypted traffic classification: datasets, representation, approaches and future thinking[C]//Proceedings of the 2024 IEEE/ACIS 24th International Conference on Computer and Information Science (ICIS). Piscataway: IEEE Press, 2024: 113-120.

[18] 张国敏, 屠智鑫, 邢长友, 等. 基于对抗样本的流量时序特征混淆方法[J]. 信息网络安全, 2024, 24(12): 1882-1895.
Zhang G M, Tu Z X, Xing C Y, et al. Traffic obfuscation method for temporal features based on adversarial example[J]. Netinfo Security, 2024, 24(12): 1882-1895.

[19] Seydali M, Khunjush F, Akbari B, et al. CBS: a deep learning approach for encrypted traffic classification with mixed spatio-temporal and statistical features[J]. IEEE Access, 2023, 11: 141674-141702.

[20] Wang W, Zhu M, Wang J L, et al. End-to-end encrypted traffic classification with one-dimensional convolution neural networks[C]//Proceedings of the 2017 IEEE International Conference on Intelligence and Security Informatics (ISI). Piscataway: IEEE Press, 2017: 43-48.

[21] Bu Z Y, Zhou B, Cheng P Y, et al. Encrypted network traffic classification using deep and parallel network-in-network models[J]. IEEE Access, 2020, 8: 132950-132959.

[22] Akbari I, Salahuddin M A, Aniva L, et al. A look behind the curtain: traffic classification in an increasingly encrypted web[J]. Proceedings of the ACM on Measurement and Analysis of Computing Systems, 2021, 5(1): 1-26.

[23] Chen Z H, Cheng G, Wei Z J, et al. Higher layers, better results: application layer feature engineering in encrypted traffic classification[C]/// Proceedings of International Conference on Wireless Algorithms, Systems, and Applications. Cham: Springer, 2022: 548-556.

[24] Xu S J, Kong K C, Jin X B, et al. Unveiling traffic paths: Explainable path signature feature-based encrypted traffic classification[J]. Computers & Security, 2025, 150: 104283.

[25] Kala J, Soukup D. Automated annotation of network traffic with data from Web browser[C]//Proceedings of 10th Prague Embedded Systems Workshop. Prague: Czech Technical University in Prague, 2022: 87-94.

[26] Torres J L G, Catania C A, Veas E. Active learning approach to label network traffic datasets[J]. Journal of Information Security and Applications, 2019, 49: 102388.

[27] V R, Poornima A S. Designing a robust network traffic annotator and classifier using active learning technique[C]//Proceedings of the 2024

Asia Pacific Conference on Innovation in Technology (APCIT). Piscataway: IEEE Press, 2024: 1-5.

[28] Cordero C G, Vasilomanolakis E, Wainakh A, et al. On generating network traffic datasets with synthetic attacks for intrusion detection[J]. ACM Transactions on Privacy and Security, 2021, 24(2): 1-39.

[29] Heng Y Q, Chandrasekhar V, Andrews J G. UTMobileNetTraffic2021: a labeled public network traffic dataset[J]. IEEE Networking Letters, 2021, 3(3): 156-160.

[30] Bomma H P. Navigating the challenges of data encryption and compliance regulations: FTP vs. SFTP[J]. International Journal of Innovative Research in Engineering & Multidisciplinary Physical Sciences, 2021, 9(5): 1-6.

[31] Faouzi J. Time series classification: a review of algorithms and implementations[C]///Proceedings of Time Series Analysis-Recent Advances, New Perspectives and Applications. London: IntechOpen, 2024: 298-332.

[32] Dai J B, Xu X L, Gao H H, et al. CMFTC: cross modality fusion efficient multitask encrypt traffic classification in IIoT environment[J]. IEEE Transactions on Network Science and Engineering, 2023, 10(6): 3989-4009.

[33] Polák M, Sedlák D, Fesl J, et al. Real data center network traffic dataset and analysis[C]//Proceedings of the 2024 IEEE 13th International Conference on Cloud Networking (CloudNet). Piscataway: IEEE Press, 2024: 1-5.

[作者简介]

付春辉（1999– ），男，山东潍坊人，信息工程大学博士生，主要研究方向为智能化网络流量分类。

杨智（1975– ），男，河南开封人，博士，信息工程大学教授、博士生导师，主要研究方向为操作系统安全、云计算安全、隐私保护。

郭渊博（1975– ），男，陕西周至人，博士，海南大学教授、博士生导师，主要研究方向为大数据安全、态势感知。

李勇飞（1998– ），男，河南开封人，信息工程大学博士生，主要研究方向为联邦学习安全。

金舒原（1974– ），女，博士，中山大学教授、博士生导师，主要研究方向为漏洞挖掘、网络攻防、人工智能安全、操作系统安全。