

北京航空航天大学学报
Journal of Beijing University of Aeronautics and Astronautics
ISSN 1001-5965,CN 11-2625/V

《北京航空航天大学学报》网络首发论文

题目：基于层级结构调制与参数耦合的加密流量分类方法
作者：郑天托，卢树华，杜翰霖
DOI：10.13700/j.bh.1001-5965.2026.0518
收稿日期：2026-07-20
网络首发日期：2026-11-06
引用格式：郑天托，卢树华，杜翰霖．基于层级结构调制与参数耦合的加密流量分类方法[J/OL]．北京航空航天大学学报.
https://doi.org/10.13700/j.bh.1001-5965.2026.0518



网络首发：在编辑部工作流程中，稿件从录用到出版要经历录用定稿、排版定稿、整期汇编定稿等阶段。录用定稿指内容已经确定，且通过同行评议、主编终审同意刊用的稿件。排版定稿指录用定稿按照期刊特定版式（包括网络呈现版式）排版后的稿件，可暂不确定出版年、卷、期和页码。整期汇编定稿指出版年、卷、期、页码均已确定的印刷或数字出版的整期汇编稿件。录用定稿网络首发稿件内容必须符合《出版管理条例》和《期刊出版管理规定》的有关规定；学术研究成果具有创新性、科学性和先进性，符合编辑部对刊文的录用要求，不存在学术不端行为及其他侵权行为；稿件内容应基本符合国家有关书刊编辑、出版的技术标准，正确使用和统一规范语言文字、符号、数字、外文字母、法定计量单位及地图标注等。为确保录用定稿网络首发的严肃性，录用定稿一经发布，不得修改论文题目、作者、机构名称和学术内容，只可基于编辑规范进行少量文字的修改。

出版确认：纸质期刊编辑部通过与《中国学术期刊（光盘版）》电子杂志社有限公司签约，在《中国学术期刊（网络版）》出版传播平台上创办与纸质期刊内容一致的网络版，以单篇或整期出版形式，在印刷出版之前刊发论文的录用定稿、排版定稿、整期汇编定稿。因为《中国学术期刊（网络版）》是国家新闻出版广电总局批准的网络连续型出版物（ISSN 2096-4188，CN 11-6037/Z），所以签约期刊的网络版上网络首发论文视为正式出版。

基于层级结构调制与参数耦合的加密流量分类方法

郑天托，卢树华✉，杜翰霖

（中国人民公安大学 信息网络安全学院，北京 100038）
✉通信作者 E-mail：lushuhua@ppsuc.edu.cn

摘要 针对现有加密流量分类方法将层级结构线索限制在静态嵌入，且粒度间序列独立建模的问题，提出一种基于层级结构调制与参数耦合的加密流量分类方法 HSMPC。层级化输入表示生成包含信息表示的层级嵌入序列与结构描述符，层级化 Mamba 块以结构类别索引可学习偏置，并加至 Softplus 变换前的离散化步长参数中，从而按结构区域调节状态保持与更新的时间尺度；参数耦合机制将字节序列编码器与流级包序列编码器中对应 Mamba 混合器的投影、卷积与状态空间参数进行耦合，使两种粒度的梯度在同一参数上联合优化。实验表明：所提方法在 CSTNET-TLS1.3、CipherSpectrum 与 LFETT2021 数据集上 3 个随机种子的加权 F1 分别达到 92.59%、94.62%与 97.36%，在 A800 GPU 上的单样本模型前向推理延迟仅为 3.5 ms。

关键词 加密流量分类；层级结构调制；Mamba；参数耦合；网络安全
中图分类号 TP393 文献标志码 文献标识码：A DOI: 10.13700/j.bh.1001-5965.2026.0518

Encrypted traffic classification based on hierarchical structure modulation and parameter coupling

ZHENG Tiantuo, LU Shuhua✉, DU Hanlin

（College of Information and Cyber Security, People's Public Security University of China, Beijing 100038, China）
✉E-mail: lushuhua@ppsuc.edu.cn

Abstract To address the problems that existing encrypted traffic classification methods confine hierarchical structural cues to static embeddings and model the sequences at different granularities independently, an encrypted traffic classification method based on hierarchical structure modulation and parameter coupling (HSMPC) was proposed. First, a hierarchical input representation generated both a hierarchical embedding sequence containing information representations and a structural descriptor, and a hierarchical Mamba block used the structural category to index a learnable bias and added the bias to the discretization step-size parameter before the Softplus transformation, thereby adjusting state-update timescales across structural regions. Second, a parameter coupling mechanism coupled the projection, convolution, and state-space parameters of corresponding Mamba mixers across the byte sequence encoder and the flow-level packet sequence encoder, enabling gradients from both granularities to jointly update the same parameters. Experimental results show that the weighted F1 scores of the proposed method, averaged over three random seeds, reach 92.59%, 94.62%, and 97.36% on the CSTNET-TLS1.3, CipherSpectrum, and LFETT2021 datasets, respectively, while its per-sample forward-pass latency on an A800 graphics processing unit (GPU) is only 3.5 ms.
Key words encrypted traffic classification; hierarchical structure modulation; Mamba; parameter coupling; cybersecurity

随着网络技术的不断进步，网络治理、业务识别、流量调度与安全防护等关键任务日益依赖高质量的流量语义理解。然而，在用户隐私保护意识增强与网络安全需求提升的双重背景下，以传输层安全协议（transport layer security，TLS）、虚拟专用网络（virtual private network，VPN）为代表的加密通信机制被广泛部署[1]；这一趋势在保障数据安全传输的同时，也使传统基于明文内容的深度包检测

收稿日期：2026-07-20
基金项目：高等学校学科创新引智基地项目（B20087）；中央高校基本科研业务费专项资金（2024JKF10）
Foundation items: The Programme of Introducing Talents of Discipline to Universities (B20087); The Fundamental Research Funds for the Central Universities (2024JKF10)
网络首发时间：2026-11-06 16:08:48 网络首发地址：https://link.cnki.net/urlid/11.2625.v.20261106.1110.001

（deep packet inspection，DPI）等方法的适用性持续减弱[2]。因此，如何在不解密载荷的条件下，通过分析有限的明文交互流量实现加密流量分类，已成为该领域亟待解决的核心问题。

传统流量分类技术依赖流量的统计特征、频域特征或流级时间特征，并配合浅层机器学习分类器完成识别；此类方法实现简单、推理高效，但其性能上界高度依赖人工特征工程，难以适应日益复杂的应用演变。近年来，深度学习的引入为加密流量分类提供了新的思路，使模型能够直接从原始字节序列、报文片段或流量图像中自动学习特征表示，降低了手工特征提取的负担。尽管现有深度学习方法在加密流量分类任务中展现出较大潜力，但其表示与建模仍存在不足。部分现有方法虽然在输入端编码字段、边界与位置等层级信息，但这些结构线索通常仅作为静态嵌入参与后续计算，缺少作用于状态转移动力学的调控。另一方面，字节序列建模与流内包序列建模通常采用独立参数化的序列混合器，不仅增加参数规模，也使两个粒度的状态转移映射相互分离。与此同时，基于自注意力机制的模型仍面临二次复杂度、显存占用和推理延迟较高的问题，难以满足在线或实时部署需求。

针对上述问题，本文提出一种基于层级结构调制与参数耦合的加密流量分类方法（hierarchical structure modulation and parameter coupling，HSMPC）。该方法基于 Mamba 模型[3-4]，以层级结构化表示驱动高效序列建模，并结合多粒度学习联合刻画字节信息、包级交互与流级语义，在保持线性复杂度的同时兼顾分类精度与推理效率。主要贡献如下：

1）构建一种层级输入表示（hierarchical input tokenization，HIT）与层级化 Mamba 块（hierarchical Mamba block，HMB）协同的状态动力学调制机制。HIT 利用结构区域类型、包内位置、流内位置和包边界建立与 Token 对齐的层级信息表示，并由其中区域类型导出逐 Token 的离散结构描述；HMB 将可学习偏置引入选择性状态空间的离散化步长，使模型形成结构条件化的状态保持与更新机制。

2）设计作用于选择性状态空间混合器的跨粒度参数耦合机制（parameter coupling across granularities，PCaG）。以字节、数据包与会话流的三段式骨干将字节序列与流内包序列的建模耦合至同一状态空间混合器中，避免重复引入独立的状态空间混合参数并实现联合优化。

3）在三个公开数据集上验证方法的分类性能与推理效率，并通过偏置量统计与置零效应、等效隐藏注意力以及耦合配置的消融实验分析层级结构线索与 PCaG 作用机制。

1 相关工作

1.1 机器学习方法

早期加密流量分类研究主要依赖人工构造特征，将分类问题转化为侧信道统计学习问题。AppScanner[5]利用流量中的 burst 模式、包长分布等统计特征训练随机森林分类器对加密流量进行识别，表明即使有效载荷已被加密，应用行为仍会在交互模式中留下可被建模的统计痕迹。该方法可解释性较强、实现成本较低，但其判别能力往往依赖于针对场景精心设计的特征空间，一旦设备、应用版本或网络环境发生变化，分类性能便容易出现显著波动。FlowLens[6]为每个流计算一种名为流标记的内存高效的包大小表示，并使用多项式朴素贝叶斯进行分类，能够以较少性能开销实现较高分类精度。但 FlowLens 仍依赖人工特征工程与传统分类器的组合，难以充分刻画加密流量中更为复杂的层级语义结构。机器学习方法实现简单、推理高效，但通常依赖经验性特征工程，且较难在统一模型中同时表达内容模式、分组关系与流级交互。

1.2 深度学习方法

深度学习方法试图绕开人工特征瓶颈，直接从原始流量中学习判别性表示。FS-Net[7]通过嵌入包长序列，利用双向门控循环单元编解码器学习并重构流序列表征，实现端到端加密流量分类。AttnLSTM[8]则将加密流量视作时序输入，在长短期记忆网络基础上引入注意力机制，以突出对分类更关键的时序片段。此类方法已经显著优于传统特征工程路线，但在长序列效率和跨样本迁移性方面仍存在局限。

近年来，以基于 Transformer 的双向编码器表征技术（bidirectional encoder representations from

Transformers，BERT）[9]为代表的自然语言处理预训练范式，以及视觉 Transformer[10]和掩码自编码器（masked autoencoder，MAE）[11]为代表的视觉预训练范式在表示学习中得到广泛应用，并逐渐被引入加密流量分析方法。ET-BERT[12]验证上下文化 datagram 表示与大规模预训练对于加密流量分类的显著价值。YaTC[13]引入 MAE 的掩码重建思想，通过字节矩阵块拆分与掩码块重建捕获流量相关性。TrafficFormer[14]进一步基于掩码突发建模和同源同向流任务预训练学习流量语义，并结合字段增强微调实现少标注条件下的高精度分类；BTRFormer[15]则引入块级流量表示与层级建模，强化包内、块间与包间关系学习。MIETT[16]进一步以多实例学习视角将每个数据包视作流袋中的独立实例，通过两级注意力联合建模 Token 级与包级依赖，以增强对流时序结构与流间差异的刻画。上述工作共同表明，层级结构化表示与自监督预训练是提升加密流量分类能力的重要方向。与上述偏重单一字节模态的工作不同，MM4flow[17]将流划分为载荷字节流与包长序列两个模态分别预训练，并以跨注意力机制实现下游任务的模态融合，显著缓解加密隧道等场景下单一模态信息失效的瓶颈。在高效序列建模方面，NetMamba[18]以状态空间模型（state space model，SSM）替代 Transformer 的自注意力计算，验证线性复杂度序列建模在网络流量任务中兼顾效率与准确率的可行性。NetMamba+[19]进一步在 Mamba 架构上引入多模态流量表示，联合建模原始报文字节、包大小和包间到达时间等互补特征，并通过多模态重构任务学习通用流量表示，在提升流量分类性能的同时保持较高的推理效率。

2　本文方法

HSMPC 的整体工作流程如图 1 所示。原始网络流量数据在预处理阶段经由无关协议清洗、匿名化处理、会话切分后进行层级化输入表示供预训练与微调使用。在预训练阶段，模型基于大规模无标签流量数据进行层级结构与字节表示驱动的自监督重建学习任务，随后将预训练编码器权重迁移到微调阶段。在微调阶段，模型对层级化输入数据进行多粒度表征学习，并将流级包序列编码器输出的分类标记（classification token，CLS）表征输入多层感知机（multilayer perceptron，MLP）分类头，得到预测结果。

具体地，HSMPC 以①HIT 提供结构线索，随后由②HMB 实现的编解码器将来自 HIT 的结构信息转换为对离散化步长进行结构条件化调制。在微调阶段利用③PCaG 的混合器统一多尺度算子，利用不同粒度信息进行流量建模，进而实现多粒度联合优化。与现有研究相比，主要区别有三：其一，ET-BERT[12]、YaTC[13]等以平铺字节序列进行掩码重建，结构信息仅隐式存在，未被显式建模与利用；本方法不仅通过 HIT 在输入端注入结构区域、边界与层级位置，更通过 HMB 将其作用于选择性扫描的离散化步长，把结构线索从被动嵌入升级为对状态递归的主动控制。其二，MIETT[16]虽引入层级化与多实例建模，但以自注意力实现、复杂度与参数较高；本文以线性复杂度的选择性状态空间承载层级化与多粒度建模，并通过 PCaG 在不额外引入第二套混合器参数的前提下完成跨粒度联合优化。其三，MM4flow[17]以多模态大模型换取性能，其参数高出本文约两个数量级；而 HSMPC 以 1.91×10^{6} 平均微调参数量的单模态轻量化设计，在多数数据集上取得相当或更优的性能与显著更低的推理延迟。

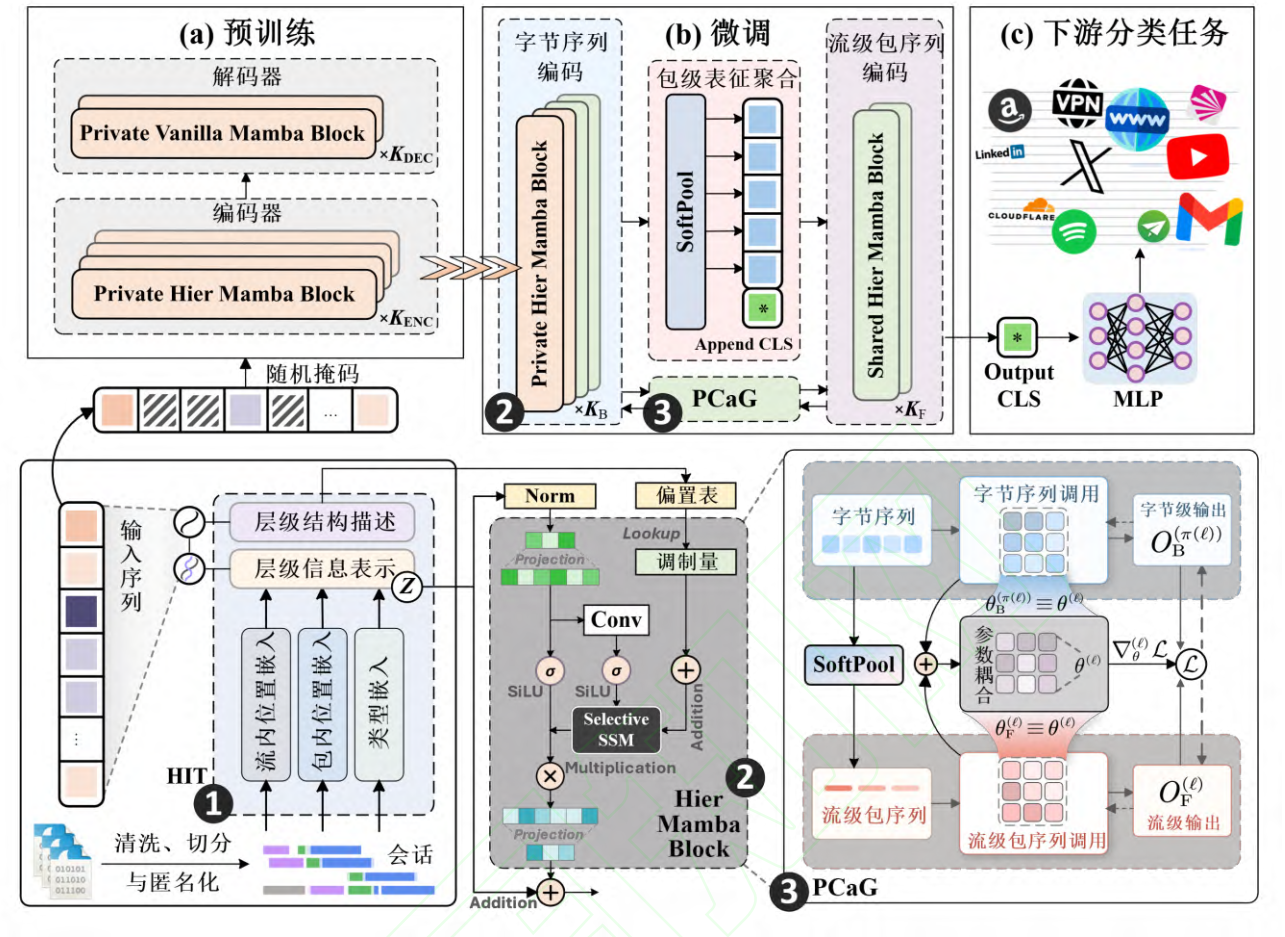


图 1 HSMPC 方法流程
Fig.1 HSMPC method flowchart

2.1 数据预处理

对于给定会话流，其内部数据包共享的、具有连续演化关系的传输控制协议（transmission control protocol，TCP）序列号与确认号，若将同一会话中的数据包或单向流划入不同子集，该特异性模式会造成严重的数据泄露[20]。同时已有研究指出，若训练过程中数据暴露明文服务器名称指示字段（server name indication，SNI）[21-22]，模型可能直接利用该字段所携带信息形成捷径特征，从而削弱对真实流量模式的建模能力与泛化能力[23]。为避免数据泄露与捷径特征，并保持与对比方法一致的输入匿名化方案，HSMPC 统一遵循如下数据处理规范：预处理首先排除非互联网协议（internet protocol，IP）数据包及以太网头部，并以完整双向会话为最小样本单位进行训练集、验证集与测试集的互斥拆分，随后对样本中 IP 地址、端口及 SNI 字段进行置零操作。默认输入保留 TCP 序列号、确认号、IP 标识与校验和；其中序列号与确认号仅在同一会话内部连续演化，采用互斥拆分以避免该模式跨子集泄露。

针对清洗后单个会话流中前 M 个数据包，保留头部字节预算 N_{h} 与有效载荷字节预算 N_{p} 并对超出或不足预算的部分进行截断或填充，从而将第 i 个数据包统一表示为长度 $N_{\mathrm{h}}+N_{\mathrm{p}}$ 的定长字节序列 $\boldsymbol{d}_i$，按时间顺序拼接为一维字节序列 $\boldsymbol{l}$：

$$\boldsymbol{l}=\left[\boldsymbol{d}_1;\boldsymbol{d}_2;\ldots;\boldsymbol{d}_M\right]\in\mathbb{R}^{L_{\mathrm{b}}} \tag{1}$$

式中：$L_{\mathrm{b}}=M\times\left(N_{\mathrm{h}}+N_{\mathrm{p}}\right)$。

设切分步幅为 L_{s}，且实验参数严格满足 $L_{\mathrm{s}}\,|\,L_{\mathrm{b}}$ 的情况下，字节序列 $\boldsymbol{l}$ 可被无重叠地划分为 $L_{\mathrm{b}}/L_{\mathrm{s}}$

个 Stride，记为 N_{s}。记第 t 个 Stride 为 $\boldsymbol{s}_t$，通过可学习的投影矩阵 $\boldsymbol{W}_{\mathrm{e}} \in \mathbb{R}^{L_{\mathrm{s}} \times D_{\mathrm{ENC}}}$ 映射到 D_{ENC} 维隐空间，得到 $\bar{\boldsymbol{z}}_t = \boldsymbol{s}_t \boldsymbol{W}_{\mathrm{e}} + \boldsymbol{b}_{\mathrm{e}} \in \mathbb{R}^{D_{\mathrm{ENC}}}$，记为 Stride Token，拼接形成扁平输入序列 $\bar{\boldsymbol{Z}}$：

$$\bar{\boldsymbol{Z}} = \left[\bar{\boldsymbol{z}}_1; \bar{\boldsymbol{z}}_2; \ldots; \bar{\boldsymbol{z}}_{N_{\mathrm{s}}}\right] \in \mathbb{R}^{N_{\mathrm{s}} \times D_{\mathrm{ENC}}} \tag{2}$$

2.2 层级化输入表示

经过数据预处理后，进一步在输入层注入流量的层级结构信息。设 $\left(N_{\mathrm{h}} + N_{\mathrm{p}}\right) / L_{\mathrm{s}}$ 为 T_{PKT}，$N_{\mathrm{h}} / L_{\mathrm{s}}$ 为 T_{h}，表示单个数据包与头部对应的 Token 数量，实验参数需满足 $L_{\mathrm{s}} \mid \left(N_{\mathrm{h}} + N_{\mathrm{p}}\right)$ 且 $L_{\mathrm{s}} \mid N_{\mathrm{h}}$。为使模型能够在仅引入极小额外序列长度与计算开销的前提下获得更细粒度的层级位置与结构信息，本方法借鉴层次化位置编码[24]与 BERT 式分隔符与类型嵌入思想[9,16]，引入结构类型嵌入、分解式位置嵌入与特殊标记嵌入，以增强模型对流内局部结构与边界的感知能力。

对于 Stride Token $\bar{\boldsymbol{z}}_t$，做流内位置嵌入与包内位置嵌入分别为 $\boldsymbol{e}_i^{\mathrm{flow}} \in \mathbb{R}^{D_{\mathrm{ENC}}}$、$\boldsymbol{e}_j^{\mathrm{pkt}} \in \mathbb{R}^{D_{\mathrm{ENC}}}$，得到分解式位置嵌入表示，并进一步依据 Token 所处结构区域引入结构类型嵌入 $\boldsymbol{e}_t^{\mathrm{type}}$，包含头部区域（header，HD）、载荷区域（payload，PL）与连续填充字节（padding，PAD）；在自然包边界处添加可学习的包边界标记 $\boldsymbol{\sigma}$（separator，SEP）及其位置嵌入 $\boldsymbol{e}^{\mathrm{sep}}$，则增强后的层级结构化 Token 表示为

$$\begin{cases} \hat{\boldsymbol{z}}_t = \bar{\boldsymbol{z}}_t + \boldsymbol{e}_i^{\mathrm{flow}} + \boldsymbol{e}_j^{\mathrm{pkt}} + \boldsymbol{e}_t^{\mathrm{type}} \\ \hat{\boldsymbol{\sigma}} = \boldsymbol{\sigma} + \boldsymbol{e}^{\mathrm{sep}} \end{cases} \tag{3}$$

记第 i 个数据包对应的增强 Token 子序列为 $\hat{\boldsymbol{Z}}_i = [\hat{\boldsymbol{z}}_{i,1}; \ldots; \hat{\boldsymbol{z}}_{i,T_{\mathrm{PKT}}}]$，因此 HIT 的有效输入序列为 $\boldsymbol{Z}$：

$$\boldsymbol{Z} = \left[\hat{\boldsymbol{Z}}_1; \hat{\boldsymbol{\sigma}}; \hat{\boldsymbol{Z}}_2; \ldots; \hat{\boldsymbol{\sigma}}; \hat{\boldsymbol{Z}}_M\right] \in \mathbb{R}^{\left(N_{\mathrm{s}} + M - 1\right) \times D_{\mathrm{ENC}}} \tag{4}$$

除连续的层级嵌入序列 $\boldsymbol{Z}$ 外，HIT 为每个序列位置生成一个逐位置对齐的结构类型描述符，记为 $S_t \in \{\mathrm{HD}, \mathrm{PL}, \mathrm{PAD}, \mathrm{SEP}\}$。

2.3 层级化 Mamba 块

如果输入嵌入中的结构线索缺乏对状态递归的直接作用，其影响可能在长序列传播过程中逐渐减弱。仅在输入端加入结构类型嵌入，并不能约束选择性状态空间显式地依据结构类型调整状态保持行为。因此，本方法提出 HMB，利用与输入序列逐位置对齐的离散结构类别 S_t 查询可学习结构偏置表 $\boldsymbol{E}_{\Delta}$，并将所得逐通道偏置加入 Softplus 变换前的离散化步长预激活，从而直接调节旧状态保持与当前状态写入之间的时间尺度。

具体地，对 HMB 输入序列的第 t 个位置，令 $\boldsymbol{x}_t$ 表示其经输入投影与因果卷积后送入步长分支的特征；对通道 d，记 $q_{t,d}$ 为 $\boldsymbol{x}_t$ 依次经低秩投影与步长投影后在该通道上的分量，则原始 Mamba 的离散化步长为

$$\Delta_{t,d} = \mathrm{Softplus}\left(q_{t,d} + b_{\Delta,d}\right) \tag{5}$$

HMB 根据 S_t 查询结构偏置表 $\boldsymbol{E}_{\Delta}$，并将对应的逐通道偏置加入步长预激活，得到结构感知的离散化步长：

$$\Delta_{t,d} = \mathrm{Softplus}\left(q_{t,d} + b_{\Delta,d} + \boldsymbol{E}_{\Delta}\left[S_t, d\right]\right) \tag{6}$$

对于通道维 d 和状态维 n，其离散状态保持系数为

$$\overline{A}_{t,d,n} = \exp\left[-\Delta_{t,d} \exp\left(A_{\log,d,n}\right)\right] \tag{7}$$

由于 Mamba 采用稳定的对角状态参数化，因此对局部固定的 $q_{t,d}$、$A_{\log,d,n}$ 和通道 d，正向结构偏置增大离散化步长，从而减小状态保持系数，使旧状态衰减加快；反之，负向结构偏置产生相反作用。因此，HMB 能够依据 Token 的结构类别自适应地学习差异化的状态更新时间尺度。

2.4 预训练

预训练方案总体沿用 NetMamba 的掩码自编码框架。以 0.9 的掩码率对每个样本的 Stride Token 进

行随机掩码，随后利用可学习的 Mask Token 和解码器位置编码恢复完整序列，并重建被掩码 Token 对应的原始字节，损失函数为仅在掩码位置计算的均方误差。

与 NetMamba 不同，本方法采用 4 层、隐藏维度为 256 的 HMB 编码器，引入分解式位置嵌入，并依据可见 Token 的结构类别查询偏置表，对状态空间离散化步长进行调制；由于解码器不受结构条件调制，因此仍选用 2 层、隐藏维度为 128 的普通 Mamba 块（vanilla Mamba block，VMB）构成。

2.5 多粒度微调

2.5.1 字节序列编码

针对式（4）的层级化表示序列，将其输入全部 K_{B} 个 HMB 构成的编码器，可得字节序列的全局建模结果 $\boldsymbol{Z}^{(\xi)}$，其中，$\xi\in\{1,\ldots,K_{\mathrm{B}}\}$：

$$\boldsymbol{Z}^{(\xi)}=\mathrm{HierMambaEncoder}^{(\xi)}\left(\boldsymbol{Z}^{(\xi-1)}\right) \tag{8}$$

2.5.2 流级包序列编码

从 $\boldsymbol{Z}^{(\xi)}$ 提取 N_{s} 个 Stride Token，对其进行 SoftPool[25]包级聚合。汇聚全部包级表征后，将同一会话流中的 M 个包表征按时序排列并在尾部，追加可学习的流级 CLS，并进行流级位置编码构造流级包序列输入 $\boldsymbol{G}^{(0)}\in\mathbb{R}^{(M+1)\times D_{\mathrm{ENC}}}$。

随后，通过 K_{F} 个 HMB 构成的流级包序列编码器建模包间依赖，其中，$\ell\in\{1,\ldots,K_{\mathrm{F}}\}$：

$$\boldsymbol{G}^{(\ell)}=\mathrm{HierMambaEncoder}^{(\ell)}\left(\boldsymbol{G}^{(\ell-1)}\right) \tag{9}$$

最终取流级 CLS 对应的输出作为流级表征，输入 MLP 分类头得到类别预测，并采用标签平滑系数为 0.1 的交叉熵损失 $\mathcal{L}$ 进行端到端优化，以完成下游分类任务。

需要说明的是，流级包序列编码器虽然复用 HMB 的混合器对象，但由于字节序列经 SoftPool 后仅存在包级聚合向量与尾部流级 CLS，不存在结构差异，故流级包序列调用中不施加结构调制。

2.6 跨粒度参数耦合

设字节级序列编码器与流级包序列编码器分别包含 K_{B} 和 K_{F} 个块，使用 ξ 和 ℓ 分别表示两种粒度下的块序号。定义映射：

$$\pi:\{1,\ldots,K_{\mathrm{F}}\}\to\{1,\ldots,K_{\mathrm{B}}\},\quad \xi=\pi(\ell) \tag{10}$$

表示流级第 ℓ 个块与字节级第 ξ 个块建立耦合关系。PCaG 施加参数耦合约束：

$$\theta_{\mathrm{B}}^{(\pi(\ell))}\equiv\theta_{\mathrm{F}}^{(\ell)}\equiv\theta^{(\ell)},\qquad \ell=1,\ldots,K_{\mathrm{F}} \tag{11}$$

式中：$\theta_{\mathrm{B}}^{(\pi(\ell))}$、$\theta_{\mathrm{F}}^{(\ell)}$ 与 $\theta^{(\ell)}$ 分别为字节级待耦合参数、流级待耦合参数及耦合后参数。

本方法采用末端耦合策略，即

$$\pi(\ell)=K_{\mathrm{B}}-K_{\mathrm{F}}+\ell \tag{12}$$

参数耦合仅作用于指定的序列混合器参数 θ，具体包含输入投影矩阵 $\boldsymbol{W}_{\mathrm{in}}$、深度可分离因果卷积核 $\boldsymbol{W}_{\mathrm{conv}}$ 及其偏置 $\boldsymbol{b}_{\mathrm{conv}}$、低秩投影矩阵 $\boldsymbol{W}_{\mathrm{x}}$、步长投影 $\boldsymbol{W}_{\Delta}$ 及其偏置 $\boldsymbol{b}_{\Delta}$、状态矩阵参数 $\boldsymbol{A}_{\log}$、连跳系数 $\boldsymbol{D}$ 与输出投影矩阵 $\boldsymbol{W}_{\mathrm{out}}$，即单个选择性状态空间混合器的全部可学习参数；各块的均方根层归一化参数、流级位置编码、流级 CLS 与分类头不参与耦合，两次调用的归一化输出、激活值与递归状态等相互独立。前向传播阶段，耦合参数分别参与字节级与流级算子调用；反向传播阶段，来自两个粒度的上游梯度经各自的局部雅可比转置映射至共享参数空间，并在参数节点处累加，形成用于参数更新的联合梯度：

$$\nabla_{\theta}^{(\ell)}\mathcal{L}=\left(\frac{\partial\boldsymbol{O}_{\mathrm{B}}^{(\pi(\ell))}}{\partial\theta^{(\ell)}}\right)^{\mathrm{T}}\frac{\partial\mathcal{L}}{\partial\boldsymbol{O}_{\mathrm{B}}^{(\pi(\ell))}}+\left(\frac{\partial\boldsymbol{O}_{\mathrm{F}}^{(\ell)}}{\partial\theta^{(\ell)}}\right)^{\mathrm{T}}\frac{\partial\mathcal{L}}{\partial\boldsymbol{O}_{\mathrm{F}}^{(\ell)}} \tag{13}$$

式中：$\boldsymbol{O}_{\mathrm{B}}^{(\pi(\ell))}$ 与 $\boldsymbol{O}_{\mathrm{F}}^{(\ell)}$ 分别为耦合后混合器在字节序列调用与流级包序列调用中的输出。

从优化角度看，参数耦合即对假设空间的一种结构化限制。当不同粒度的梯度方向一致或近似正

交时，联合梯度在共享参数上累加，两任务互不干扰；当二者显著冲突时，共享参数被迫在两种变换间折中。

3　实验与结果分析

3.1 数据集

预训练阶段的语料由大规模无标签流量构成，其来源包括 WIDE 项目 MAWI 骨干网流量存档[26]中 2026 年 1 月 1 日、11 日与 21 日三个采集日的流量数据，以及 Browser[27]与 Proxy2023[28]数据集。在下游任务微调阶段，采用 CSTNET-TLS1.3[12]与 CipherSpectrum[23]发布的均衡混合密码套件加密的会话数据集开展 TLS1.3 加密流量分类任务的训练，二者分别包含 120 个类别共 46372 个会话与 41 个类别共 41205 条会话（CipherSpectrum 每类 1005 条样本）；采用 LFETT2021[29]数据集开展隧道加密流量识别任务的训练，该数据集包含 38 个类别共 95461 条会话，其中覆盖基于主流代理协议 Shadowsocks-R 与 VMess 构建的加密隧道流量。

3.2 实验设置

本文分类精度对比与消融实验均在 Ubuntu 22.04 操作系统下完成，软件环境为 Python 3.10.13、PyTorch 2.1.1 和 CUDA 12.1。硬件平台配置为 NVIDIA RTX 4090 GPU（24 GB 显存）以及 16 vCPU Intel Xeon Platinum 8352V 处理器。推理效率对比实验则在 NVIDIA A800 GPU（80 GB 显存）硬件平台进行。

预训练阶段掩码率设为 0.9，Batch Size 设为 128，基础学习率设为 1.5×10^{-4}，训练步数为 2×10^{5}。微调阶段使用 AdamW 优化器，Batch Size 设为 64，训练轮次为 120，基础学习率设为 5×10^{-4}，权重衰减系数为 0.05。各微调数据集划分比例为 8:1:1，下游任务微调时保存验证集上最佳准确率的检查点，随后使用测试集进行评估。为保证实验结果的稳定性，本方法主要结果、关键消融实验与部分探究实验均使用三个随机种子（0，1，2）进行独立微调，报告均值与标准差。

3.3 评估指标

为全面评估模型性能，本文分别从分类性能与模型效率两个维度进行实验分析。在分类性能方面，采用准确率（accuracy，AC）、加权精确率（weighted precision，wPR）、加权召回率（weighted recall，wRC）和加权 F1 分数（weighted F1 score，wF1）作为评价指标，均以百分数表示。具体公式如下：

$$\begin{cases} \eta = \dfrac{1}{N_{\text{sam}}}\displaystyle\sum_{c=1}^{C} N_{\text{TP},c} \\ P_{\text{w}} = \dfrac{1}{N_{\text{sam}}}\displaystyle\sum_{c=1}^{C} N_{\text{sup},c}\dfrac{N_{\text{TP},c}}{N_{\text{TP},c}+N_{\text{FP},c}} \\ R_{\text{w}} = \dfrac{1}{N_{\text{sam}}}\displaystyle\sum_{c=1}^{C} N_{\text{sup},c}\dfrac{N_{\text{TP},c}}{N_{\text{TP},c}+N_{\text{FN},c}} \\ F_{\text{w}} = \dfrac{1}{N_{\text{sam}}}\displaystyle\sum_{c=1}^{C} N_{\text{sup},c}\dfrac{2N_{\text{TP},c}}{2N_{\text{TP},c}+N_{\text{FP},c}+N_{\text{FN},c}} \end{cases} \tag{14}$$

式中：η、P_{w}、R_{w}与F_{w}分别对应指标 AC、wPR、wRC 与 wF1；C为类别数，N_{sam}为参与评价的样本总数，$N_{\text{sup},c}$为第c类样本的支持度；$N_{\text{TP},c}$、$N_{\text{FP},c}$与$N_{\text{FN},c}$分别为第c类的真阳性、假阳性与假阴性数量。由于加权召回率以各类支持度为权重，且$N_{\text{TP},c}+N_{\text{FN},c}=N_{\text{sup},c}$，$R_{\text{w}}$与$\eta$恒等，后续实验不再报告$R_{\text{w}}$及其对应指标 wRC。

在模型效率方面，通过在不同批大小下统计推理延迟和显存占用，用于刻画在线推理场景与吞吐导向场景的可部署性。

3.4 实验结果分析

3.4.1 分类任务性能对比分析

为评估 HSMPC 在不同场景下的分类性能，选取 CSTNET-TLS1.3、CipherSpectrum 和 LFETT2021 数据集展开对比实验。所对比基线涵盖基于传统特征工程的方法（如 AppScanner）、基于序列属性的方法（如 FlowLens、FS-Net）、基于原始流量表示的方法（如 AttnLSTM、ET-BERT、NetMamba 等）、基于层级化块或词袋表示的方法（如 MIETT、BTRFormer）。为保证对比的公平性，所有预训练方法均在本文同一无标签语料上按照其公开实现重新预训练，且采用相同下游样本划分、匿名化口径与硬件设备，但输入表示与字节、数据包或流级预算均遵循各方法原设计，以避免改变其模型假设。

各方法的比较结果如表 1 所示，其中 HSMPC 在 CSTNET-TLS1.3、CipherSpectrum 和 LFETT2021 上的 wF1 分别为(92.59±0.21)%、(94.62±0.24)%和(97.36±0.13)%。综合 AC、wPR 和 wF1，9 项结果的标准差均处于 0.11%～0.24%之间，平均约为 0.19%，表明模型在所考察的随机种子下具有较小的性能波动。

基于平均微调参数量仅为 1.91×10^6 的轻量级主干，HSMPC 在 CipherSpectrum 和 LFETT2021 上取得最优分类性能，其平均 wF1 分别较次优方法 MM4flow 和 YaTC 高出 1.09%、1.13%，且较 NetMamba 在三个数据集上的平均 wF1 分别高出 2.32%、1.50%、3.36%。NetMamba+在 CSTNET-TLS1.3 上领先本方法 1.01%，但在 CipherSpectrum 和 LFETT2021 上分别落后 1.40%、13.14%。与 MM4flow 相比，其 wF1 在 CSTNET-TLS1.3 和 CipherSpectrum 上分别低 0.76%和 0.31%，在 LFETT2021 上则高出 9.53%。ET-BERT、MIETT 等重量级模型未体现稳定优势，仅 MM4flow 在 CSTNET-TLS1.3 上领先本方法 1.77%，但其平均微调参数量达到 179.41×10^6。

不同数据集对模型能力的考查各有侧重。CSTNET-TLS1.3 包含大量网站类别且分布不均，误差主要来自同一服务商旗下握手模式相近的细粒度类别。CipherSpectrum 类别近似均衡，更侧重握手结构差异。而 LFETT2021 的协议结构和载荷统计特性与两个 TLS 数据集明显不同，模型排序也随之变化。依赖局部字节共现的方法整体落后，与该任务可能更依赖跨位置关系、而较少依赖相邻字节共现的解释一致。NetMamba+和 MM4flow 在 LFETT2021 数据集的分类表现未能延续，wF1 仅为 84.22%和 74.69%，说明多模态信息融合或模型容量优势在协议结构变化后未必能稳定转化为分类性能。HSMPC 在三项指标上均保持领先，体现了其对异质流量结构的适应能力。

表 1 CSTNET-TLS1.3、CipherSpectrum与LFETT2021的比较结果

Table 1 Comparison results on CSTNET-TLS1.3, CipherSpectrum and LFETT2021

方法	参数量/10^6		CSTNET-TLS1.3			CipherSpectrum			LFETT2021		
	预训练	平均微调	AC	wPR	wF1	AC	wPR	wF1	AC	wPR	wF1
AppScanner	-	-	35.57	41.45	32.22	52.13	59.27	51.31	47.32	51.12	45.93
FlowLens	-	8.6	39.36	40.81	35.23	53.63	60.39	51.50	23.89	40.71	26.86
FS-Net	-	5.3	78.88	78.12	78.10	92.09	92.22	92.06	70.80	71.79	70.76
AttnLSTM	-	-	64.15	62.72	62.16	73.45	73.83	73.31	63.57	63.34	62.11
ET-BERT	187.4	132.1	88.92	89.78	88.89	85.00	85.30	84.76	68.14	70.46	66.55
YaTC	2.3	2.1	88.00	87.77	87.89	93.20	93.29	93.19	96.13	96.51	96.23
NetMamba	2.14	1.86	90.40	90.43	90.27	93.12	93.21	93.12	93.98	94.43	94.00
MIETT	264.8	216.9	88.71	89.22	88.74	89.32	91.09	89.63	77.50	77.86	75.34
TrafficFormer	187.4	136.4	84.84	83.71	84.35	86.91	87.24	86.91	81.09	82.73	81.92
BTRFormer	2.23	1.83	75.19	75.41	73.44	83.80	84.95	83.75	85.24	85.45	84.49
MM4flow	174.4	179.41	94.37	94.56	94.36	93.72	94.39	93.53	74.48	75.59	74.69
NetMamba+	2.96	1.89	93.63	93.75	93.60	93.18	93.51	93.22	84.28	84.31	84.22
Ours	2.17	1.91	92.65 ±0.20	92.71 ±0.19	92.59 ±0.21	94.62 ±0.24	94.67 ±0.22	94.62 ±0.24	97.34 ±0.11	97.49 ±0.15	97.36 ±0.13

LFETT2021 上的性能差异可结合表 2 会话样本的统计结果加以解释。其中，零阶熵 H_0 衡量忽略上下文时单个字节的平均不确定性，一阶条件熵 H_1 衡量已知前一字节时当前字节的剩余不确定性，熵差则反映相邻字节依赖带来的可预测性增益；熵值越接近 8，字节分布越接近均匀，而 ΔH 越小，说明相邻字节的一阶统计依赖越弱；所有熵值单位为 bit Ƀ$^{-1}$。尽管三个数据集的载荷均具有高熵，但 LFETT2021 载荷的字节更接近均匀分布，相邻字节之间保留的一阶依赖也更弱；同时，其包含 TLS 记

录头的数据包占比接近 0，难以提供稳定的握手结构锚点。对于依赖连续字节 Token 掩码预测和局部统计关联的预训练模型而言，两方面因素共同减少可供学习和迁移的局部模式。NetMamba 与 YaTC 注重数据包头部与有效载荷内部组织关系，对局部字节共现模式的依赖较低；HSMPC 则进一步利用层级结构信息调制状态递归过程，并通过 PCaG 进行多粒度联合优化，更关注头部、载荷以及跨包序列的整体组织关系。NetMamba+虽然在两个 TLS 数据集上较 NetMamba 有所提升，但其在 LFETT2021 上的 wF1 为 84.22%；该模型额外使用数据包长度与到达时间间隔模态，这一结果说明，仅增加多模态信息并不必然改善高熵隧道流量的识别效果，输入表示与模型机制之间的适配同样重要。综上，HSMPC 在三个数据集上均保持较高性能，体现出层级结构调制与轻量状态空间骨干相结合所带来的跨场景适应性。

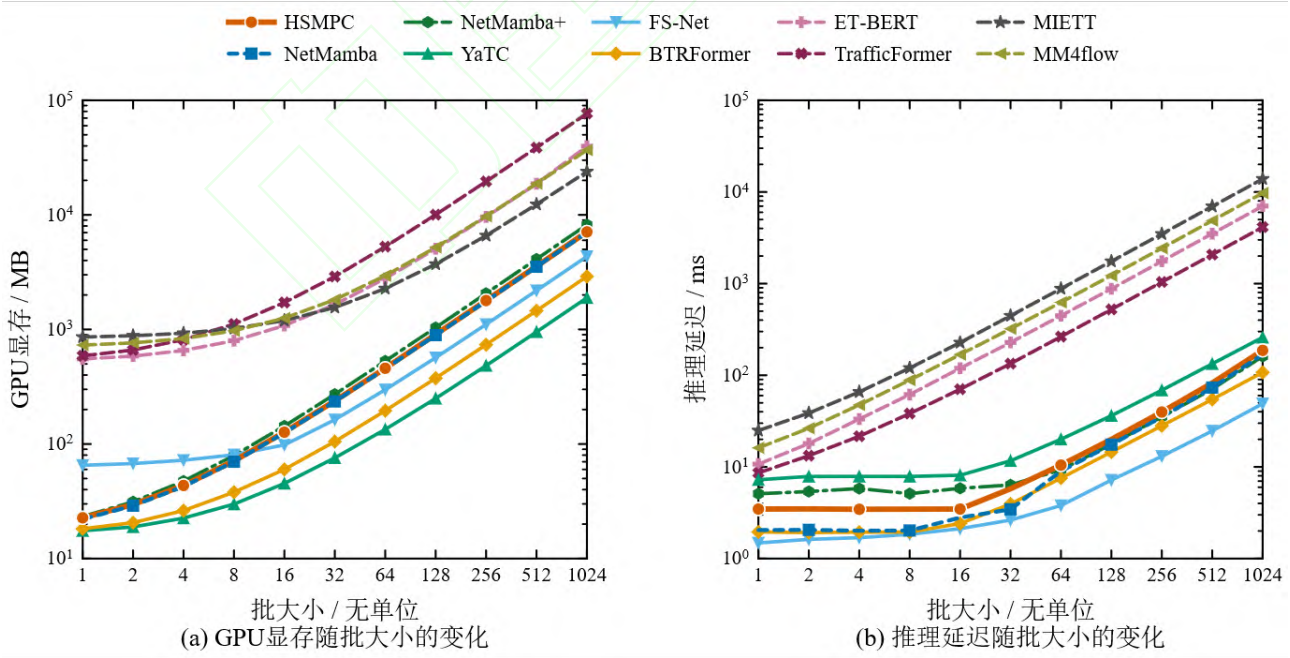
表 2 各数据集头部与载荷区域的字节可预测性统计数据

Table 2　Statistical data on byte predictability in the header and payload regions of each dataset

数据集	含 TLS 记录头数据包占比/%	区域	内容占比/%	零阶熵 H_0	一阶条件熵 H_1	熵差 ΔH
CSTNET-TLS1.3	35.7	头部	55.2	6.167	5.118	1.049
		载荷	67.0	7.924	7.716	0.208
CipherSpectrum	32.2	头部	47.5	5.896	4.564	1.332
		载荷	50.5	7.920	7.681	0.239
LFETT2021	0.4	头部	56.8	5.834	4.442	1.392
		载荷	22.9	7.993	7.922	0.071

3.4.2 推理效率对比分析

为评估所提方法的推理效率，本节从推理延迟与峰值显存占用两个维度将其与现有深度学习方法进行对比。推理延迟实验分别在单样本与不同批大小配置下开展；考虑到部分基线方法在大批次配置下使用 32 位单精度浮点时显存占用极高，推理效率对比实验统一在 A800 GPU（80 GB 显存）上进行。为消除冷启动开销对计时结果的干扰，实验均执行 5 次预热前向传播，随后取 20 次重复实验的平均值作为该批大小下的推理延迟；测试显存占用前重置显存峰值统计，并在测量前后进行 GPU 同步以消除无关影响。本节报告指标均为模型前向计算延迟与显存占用，不包含流量采集、会话构建与预处理开销，用于刻画模型在线计算的可部署性。



(a) GPU显存随批大小的变化　(b) 推理延迟随批大小的变化

图 2 不同批大小下各方法显存与推理延迟对比

Fig.2 Comparison of GPU memory and inference latency across different batch sizes

如图 2 所示，各方法在两幅双对数图上呈现出清晰的效率分层，HSMPC、NetMamba、

NetMamba+、YaTC、BTRFormer 和 FS-Net 构成轻量带，ET-BERT、TrafficFormer、MM4flow 与 MIETT 组成重量带。单样本推理时，HSMPC 仅占用 22.70 MB 显存，为重量级模型的 2.6%~4.1%。尽管显存占用差距随批大小增加有所收窄，所提方法在批大小 1024 时显存消耗仍仅为重量模型的 9.3%~29.4%，批量推理耗时为 186.74 ms，而 TrafficFormer、ET-BERT、MM4flow 和 MIETT 的推理延迟分别达到其 22.1、37.6、52.1 和 74.4 倍。批大小从 2^0 增至 2^4 时，HSMPC 的推理延迟基本恒定在 3.5 ms，并行样本数的增加未引起明显延迟增长；批大小超过 32 后，延迟曲线进入近乎理想的线性增长区，并在批大小为 256 时取得最低 0.18 ms 的单流均摊延迟。而以 ET-BERT 为代表的重量模型在由单样本转入双样本并行时延迟已从 10.73 ms 升至 18.01 ms，表明其可用于批并行均摊的剩余计算空间相对有限。

结合表 1 可知，资源开销与 HSMPC 持平或更低的基线方法，其分类性能总体较弱；但 NetMamba+与 MM4flow 均在 CSTNET-TLS1.3 数据集上取得优于 HSMPC 的分类性能。在批大小为 1024 时，NetMamba+的推理延迟为 160.77 ms，较本方法低 13.9%，但显存占用高出 15.5%，体现以适度增加显存为代价兼顾分类性能与大批量吞吐能力的特点；MM4flow 的推理延迟与显存占用分别为所提方法的 52.1 倍与 5.2 倍，其以显著更高的计算开销实现性能提升。综合表 1 与图 2 数据，尽管 HSMPC 在部分数据集分类性能略差，但整体展现出良好的精度与效率的均衡，能够以较低的显存占用与推理延迟实现与先进深度学习方法相当或更优的分类性能

3.5 消融实验

为评估所提方法中各核心组件对模型性能的贡献，本文在 CSTNET-TLS1.3、CipherSpectrum 与 LFETT2021 三个数据集上开展消融实验。各变体的结果汇总于表 3，具体改动说明如下：

w/o PCaG：保持字节序列编码、包级表征聚合与流级包序列编码路径不变，仅取消对应 Mamba 混合器之间的参数耦合，使两级调用分别采用独立参数，用于评估跨粒度参数耦合的作用。w/o HMB：保留 HIT 提供的层级输入表示，在预训练与微调阶段均将编码器中的 HMB 替换为普通 Mamba 块 VMB，即移除结构描述查询偏置表内对离散化步长的调制量，用于评估在保留 HIT 条件下 HMB 的增量贡献。w/o HIT：移除结构区域类型嵌入、分解式位置嵌入及层级结构描述。由于 HMB 失去可用于索引结构偏置的标签，该变体中的 HMB 自然退化为普通 Mamba 块，用于评估 HIT 在普通 Mamba 骨干下的增量贡献。w/ LTF：将编码器与解码器中的 Mamba 块替换为线性 Transformer（linear Transformer，LTF）[30]，用于比较不同序列建模骨干的分类性能与参数效率。w/o Pretrain 变体直接以随机初始化在各下游数据集上训练，其余配置与默认模型一致，用于评估自监督预训练本身的贡献。

表 3 消融实验结果

Table 3 Ablation experiment

消融	参数量/10^6		CSTNET-TLS1.3			CipherSpectrum			LFETT2021		
	预训练	平均微调	AC	wPR	wF1	AC	wPR	wF1	AC	wPR	wF1
Default	2.17	1.91	92.65 ±0.20	92.71 ±0.19	92.59 ±0.21	94.62 ±0.24	94.67 ±0.22	94.62 ±0.24	97.34 ±0.11	97.49 ±0.15	97.36 ±0.13
w/o PCaG	2.17	2.74	90.96 ±0.44	91.01 ±0.40	90.88 ±0.40	93.09 ±0.26	93.13 ±0.25	93.08 ±0.25	96.63 ±0.55	96.87 ±0.49	96.68 ±0.53
w/o HMB	2.17	1.91	91.34 ±0.24	91.38 ±0.24	91.22 ±0.25	92.87 ±0.30	92.93 ±0.30	92.87 ±0.33	96.98 ±0.19	97.10 ±0.18	97.00 ±0.19
w/o HIT	2.17	1.91	89.90 ±0.28	89.96 ±0.28	89.79 ±0.29	91.80 ±0.19	91.86 ±0.17	91.75 ±0.24	96.54 ±0.38	96.76 ±0.31	96.57 ±0.33
w/ LTF	3.53	3.22	87.55	87.59	87.45	88.08	85.12	84.48	95.65	95.99	95.69
w/o Pretrain	-	1.91	85.73	85.77	85.54	88.42	88.63	88.40	96.73	96.91	96.76

在架构层面，w/o PCaG 变体使微调阶段平均参数量由 1.91×10^6 增加至 2.74×10^6，增幅约 43.5%，而其在三个数据集上的 wF1 分别下降 1.71%、1.54%和 0.68%。该结果表明，单纯增加独立参数并未带来性能收益，跨粒度参数耦合则能够在压缩参数规模的同时提升分类性能。

以线性 Transformer 替换 Mamba 块的 w/ LTF 变体表现出严重的退化，尤其在 CipherSpectrum 上的 wF1 下降达 10.14%，分类性能明显劣于 Mamba 块实现的模型。该结果表明，在密文流量长序列表征任务中，选择性状态空间建模相较于线性注意力变体具有更强的判别能力和更优的参数效率。

以默认模型的多种子均值为参照，无预训练变体在 CSTNET-TLS1.3 与 CipherSpectrum 上的 wF1

分别下降 7.05%与 6.22%，而在 LFETT2021 上仅下降 0.60%。该现象可由掩码重建目标的可学习性加以解释，基于字节连续掩码预测的自监督学习，其有效性以相邻字节之间保留足够的条件可预测性为前提。通过分析表 2 中可预测性统计数据，可知其有效载荷仅占 22.9%且零阶熵与一阶条件熵均接近均匀分布，相邻载荷字节几乎不具有条件可预测性。该统计为 LFETT2021 上预训练增益较小提供一种可能解释：其载荷更接近高熵分布，基于相邻字节条件依赖的重建目标可利用的局部规律较少。

HIT 生成层级输入表示及与 Token 对齐的结构描述符，HMB 则进一步利用层级结构信息调制选择性状态空间的离散化步长，即 HIT 与 HMB 具有天然的嵌套关系，因此本节通过两组条件比较分析其增量贡献。在保留 HIT 的条件下，默认模型相较 w/o HMB 变体在 3 个数据集上的 wF1 分别提高 1.37%、1.75%和 0.36%，表明 HMB 对状态递归的结构调制在三个数据集上均能带来额外收益。同时 w/o HMB 变体相较 w/o HIT 变体在三个数据集上的平均 wF1 分别提高 1.43%、1.12%、0.43%。结合两组比较可以看出，HIT 的层级信息表示与 HMB 引入的状态动力学调制在三个数据集上均带来正向贡献，且 HIT 带来的增量均高于 HMB，说明输入侧的信息表示是性能提升的主要来源。两项改进在 LFETT2021 上的增量幅度整体小于其余数据集，说明不同数据集对层级表示和动态状态调制的敏感程度存在差异。

3.6 核心组件分析

为进一步考察 HMB 利用 HIT 提供的层级结构描述符调制状态动力学的作用特征，以及 PCaG 耦合策略对分类性能的影响，本节从状态动力学与耦合配置角度对核心组件进行分析。

3.6.1 状态动力学分析

对于每个结构类别，将偏置量在通道维取平均得到 μ，并在三个随机种子上报告均值与标准差。同时在保持输入、模型参数与其他结构偏置不变的情况下，使用置零方式分析不同结构对当前步长和状态保持行为的影响。使用 δ_{Δ} 表示离散化步长相对置零情况的相对变化率，δ_{R} 表示状态保持率相对置零情况的绝对变化量，单位均为%。置零效应仅度量推理阶段逐位置的局部通路，不能反映其沿完整序列的累积效应与训练期参数协同效应。

表 4　HMB结构偏置均值、局部置零效应及匹配冻结控制下的参数差异

Table 4　Structural mean biases, local zeroing effects and parameter differences under the matched frozen control for HMB

分析对象	统计量	B1	B2	B3	B4
HD	μ	−0.0009±0.0002	+0.0210±0.0003	+0.0260±0.0007	+0.0458±0.0017
	$\delta_{\Delta}/\delta_{\mathrm{R}}$	+1.01 / −0.073	+5.61 / −0.849	+4.07 / −0.383	+5.15 / −0.702
PL	μ	+0.0284±0.0008	+0.0521±0.0002	+0.0598±0.0007	+0.0353±0.0016
	$\delta_{\Delta}/\delta_{\mathrm{R}}$	+12.80 / −0.721	+8.96 / −1.209	+6.00 / −0.442	+4.94 / −0.415
PAD	μ	−0.0894±0.0003	−0.0095±0.0003	+0.0120±0.0002	+0.0420±0.0013
	$\delta_{\Delta}/\delta_{\mathrm{R}}$	−10.94 / +0.766	+2.40 / −0.293	+1.14 / −0.080	+4.50 / −0.403
SEP	μ	−0.0001±0.0004	−0.0009±0.0001	−0.0017±0.0004	−0.0022±0.0011
	$\delta_{\Delta}/\delta_{\mathrm{R}}$	+0.13 / −0.006	−0.01 / +0.000	−0.02 / +0.001	+0.02 / −0.002
非偏置混合器参数	$d_{\mathrm{L2}}^{\mathrm{mixer}}$	5.23	7.19	9.79	16.11

表 4 报告最终偏置均值与置零效应，呈现出稳定的结构类别与编码深度调制模式。PL 在各层保持一致方向，HD 由浅层的弱调节转为中后层的明确调节，PAD 则随深度发生方向转换，表明 HMB 未退化为对全部 Token 施加相同时间尺度。由于通道间正负变化可能在行均值中抵消，具体记忆效应可通过置零效应进行分析。HD 和 PL 整体表现为加快状态更新，说明模型倾向于在有效内容位置更快吸收当前输入；PAD 在浅层减少状态更新、在深层转为较快更新，可能反映浅层首先抑制连续填充造成的无效写入，而深层状态已混合上下文后采用不同的策略。部分单元中偏置均值与聚合步长方向不一致，可能来自通道异质性、Softplus 非线性及状态参数加权的作用。SEP 偏置行的局部置零效应接近零，说明模型未通过该偏置行实施统一的包间时间尺度调制。完整移除 SEP 的变体在两个 TLS 数据集上表现与默认模型持平（wF1 分别提升 0.14%与 0.04%），而在 LFETT2021 数据集下降 1.15%，表明其相关输入路径可能参与分类。需要指出的是，结构偏置的数值幅度及逐位置置零效应虽较小，但

这并不意味着其作用有限。推理过程中，调制会沿序列传递并在网络深度上逐层复合，因此局部扰动可通过递归状态更新被持续累积。

为进一步考察结构偏置在微调阶段保持可用所伴随的训练结果变化，本文从同一预训练检查点出发，采用相同随机种子与微调配置训练偏置冻结控制，并在加载预训练权重后将结构偏置置零且保持冻结。该控制保留 HMB 的其余计算路径，仅关闭微调阶段的显式结构偏置通路，因此不同于表 3 中将 HMB 替换为普通 Mamba 块的 w/o HMB 变体。比较两个最终检查点可见，HMB 与偏置冻结控制的非偏置混合器参数相对 L2 距离由 B1 的 5.23%增加至 B4 的 16.11%，表明匹配微调后两条训练轨迹形成了随深度增加的参数差异。

尽管 Mamba 并不显式构造注意力矩阵，但将选择性状态空间的递归逐通道展开后，其核函数 $K_{t,s}$ 随输入变化，起到与注意力权重相当的数据依赖加权作用，可视为一种等效隐藏注意力。因此 $|K_{t,s}|$ 刻画了查询位置 t 对源位置 s 的等效依赖强度，其在全部位置对上的分布构成一个下三角结构的等效隐藏注意力矩阵[31]。由于该分析量同时由离散化步长、输入相关投影和其余已训练参数共同决定，相关差异只用于描述不同模型的整体动力学。

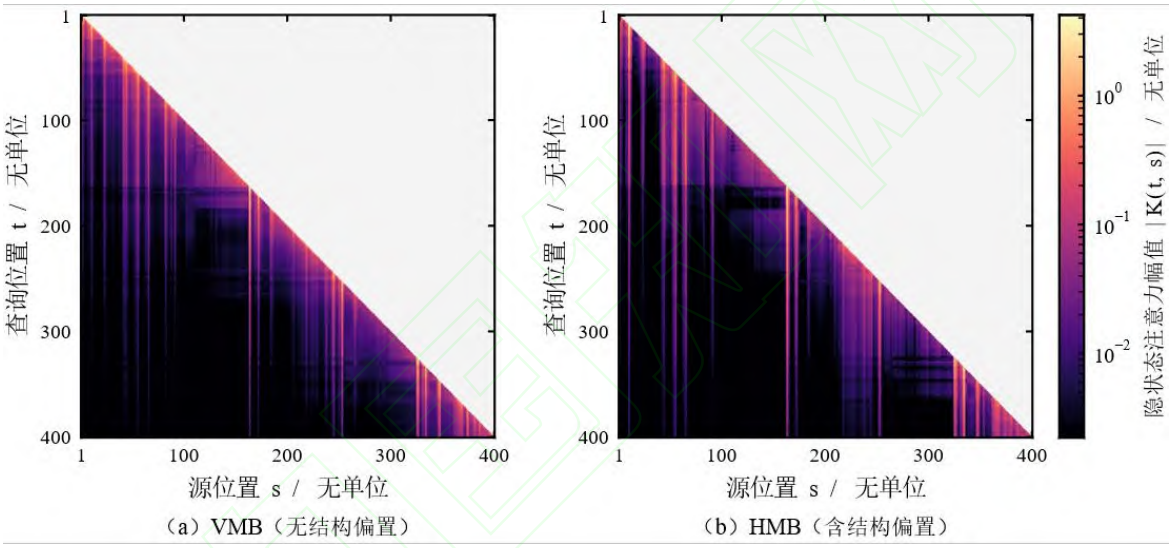


图 3 VMB 与 HMB 在 CipherSpectrum 测试集上的等效隐藏注意力
Fig.3 Equivalent hidden attention of VMB and HMB on CipherSpectrum test set

图 3 展示 VMB 与 HMB 在 CipherSpectrum 测试集上的等效依赖分布。VMB 的响应集中在对角线附近，并随查询位置与源位置之间的距离增加而整体衰减。而 HMB 在保留清晰的主对角线邻域响应外，在若干源位置形成跨越较长查询范围的高响应带。该变化并非针对全部远距离位置的均匀增强，而是将部分等效依赖质量集中到特定源位置，说明 HMB 能够在维持局部响应的同时形成更具选择性的位置相关长程依赖分布。

3.6.2 PCaG 耦合配置分析

为考察 PCaG 参数耦合在不同位置的作用及末端内部各耦合的贡献，本节在 CipherSpectrum 数据集上进行不同耦合窗口位置与窗口内组合的对照实验。字节序列编码器包含 4 个依次连接的 HMB，记为 Byte_1 至 Byte_4；流级包序列编码器包含 2 个 HMB，记为 Flow_1 与 Flow_2；不同配置的耦合映射关系位于表中。未建立耦合关系的混合器采用独立参数并由相同预训练参数初始化，所有配置均采用相同的预训练检查点、数据划分与微调设置。表 5 报告参数量与分类性能，并以 wF1 增益表示相对无耦合配置的变化量。

表 5　CipherSpectrum数据集上不同PCaG配置消融结果

Table 5　Ablation results for different PCaG configurations on CipherSpectrum

耦合配置	耦合映射		参数量/10^6	AC	wPR	wF1	wF1 增益
无耦合	—		2.68	93.09±0.26	93.13±0.25	93.08±0.25	—
前端窗口完整耦合	$Byte_{1,2}$	$Flow_{1,2}$	1.80	94.09±0.09	94.11±0.09	94.08±0.08	+1.00
中间窗口完整耦合	$Byte_{2,3}$	$Flow_{1,2}$	1.80	93.48±0.22	93.55±0.21	93.47±0.24	+0.39
末端窗口上位耦合	$Byte_4$	$Flow_2$	2.24	94.12±0.19	94.14±0.19	94.11±0.19	+1.03
末端窗口下位耦合	$Byte_3$	$Flow_1$	2.24	93.74±0.21	93.78±0.20	93.74±0.21	+0.66
末端窗口完整耦合	$Byte_{3,4}$	$Flow_{1,2}$	1.80	94.62±0.24	94.67±0.22	94.62±0.24	+1.54

分析表 5 可知，不同耦合配置相较于无耦合基线的 wF1 增益介于 0.39%至 1.54%，其中末端窗口完整耦合的增益最大，中间窗口最小，表明参数耦合在各候选位置上均带来正向收益，而收益幅度取决于耦合约束所处的编码位置。而末端窗口内部两个耦合对应项的贡献存在不对称性，上位耦合项单独使用时增益为 1.03%，高于下位耦合项的 0.66%；完整窗口的增益为 1.54%，其平均结果高于任一单项，说明在当前设置下同时保留两个末端耦合对应项更为有利。

为减少耦合位置选择对逐一试验的依赖，本节在统一预训练检查点以任务信息充分度、耦合更新相容度与短程参数分离度三个方面考察候选耦合窗口。三项指标分别采用冻结编码器的线性探针性能、同一次耦合更新对两条路径目标的兼顾程度，以及相同初始化下独立短程更新形成的相对参数间距衡量。图 4 结果表明，随着耦合窗口位置加深，任务信息充分度与耦合更新相容度总体提高，参数间距逐步减小；其中末端窗口在三项指标上形成一致优势，并与其完整耦合取得最高 wF1 的实验现象相互印证。三项指标仅作为当前架构下缩小候选范围的启发式诊断量：对于不同深度或结构的模型，可优先在验证集上考察任务信息较充分、更新相容度较高且参数间距较小的连续窗口，其适用性仍需通过完整验证实验确认。

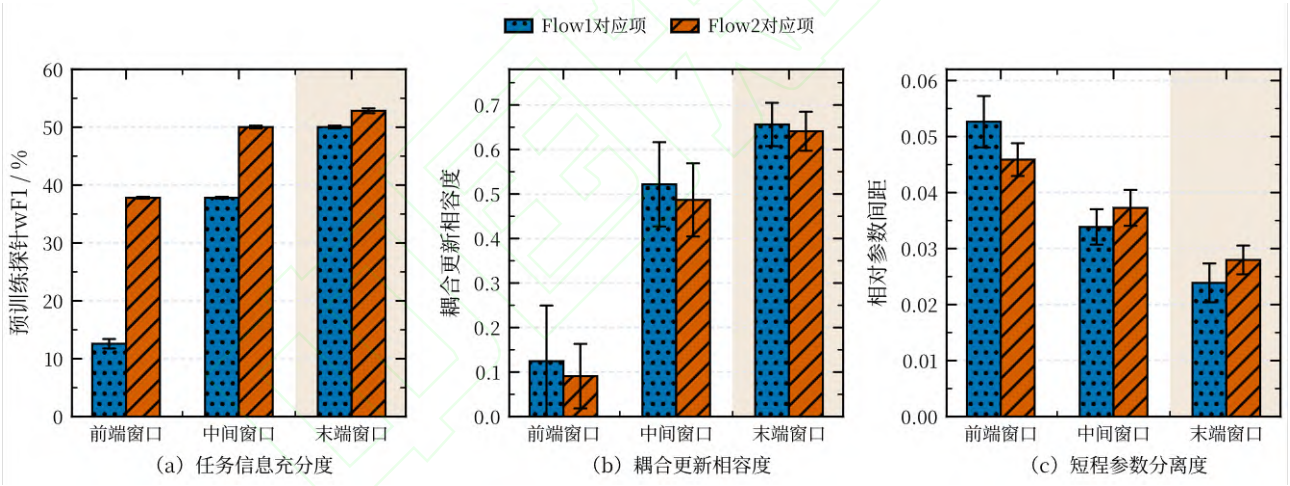


图 4 PCaG 耦合窗口筛选指标

Fig.4 Metrics for PCaG coupling selection

3.7 鲁棒性实验

3.7.1 特定字段依赖实验

为排除模型利用特定字段形成分类捷径的可能性，本节在现有匿名化方案的基础上进一步将 IP 标识和校验和置零，并重新微调 YaTC、NetMamba 和 HSMPC，结果见表 6。采用更严格的匿名化方案后，YaTC 的三项指标绝对变化均不超过 0.18%，NetMamba 的三项指标下降 0.28%至 0.64%；HSMPC 在两个 TLS 数据集上的 wF1 降幅均不超过 0.20%，而在 LFETT2021 上的 wF1 下降 0.94%，表明 HSMPC 在该数据集上的分类结果对上述字段更为敏感。

表 6 严格匿名化后的分类性能

Table 6 Classification performance following strict anonymization

方法		CSTNET-TLS1.3			CipherSpectrum			LFETT2021		
		AC	wPR	wF1	AC	wPR	wF1	AC	wPR	wF1
YaTC	移除字段后	87.82	87.91	87.71	93.25	93.28	93.22	96.10	96.35	96.16
	Δ	−0.18	+0.14	−0.18	+0.05	−0.01	+0.03	−0.03	−0.16	−0.07
NetMamba	移除字段后	89.91	90.01	89.89	92.84	92.90	92.82	93.51	93.79	93.57
	Δ	−0.49	−0.42	−0.38	−0.28	−0.31	−0.30	−0.47	−0.64	−0.43
HSMPC	移除字段后	92.52	92.63	92.44	94.42	94.48	94.42	96.33	96.71	96.42
	Δ	−0.13	−0.08	−0.15	−0.20	−0.19	−0.20	−1.01	−0.78	−0.94

3.7.2 丢包扰动实验

实际加密流量监测场景中，由于链路镜像或原始通信过程可能产生传输丢包，导致捕获流量不完整。对于依赖报文字节及跨包上下文进行判别的模型，不仅会减少可用信息，还可能破坏握手过程、包间关联及流级结构，从而造成模型输入分布偏移。真实链路上的丢包在采集侧存在两种不同的模式：丢失数据包被传输层重传机制恢复，报文内容最终仍被采集但观测序列中出现重复与位移，称为传输层恢复型丢包（transport-layer recovered loss，TRL）；数据包在采集点之前丢失且恢复过程不被观测，丢包表现为报文缺失与内容损失，称为 Gilbert–Elliott 丢包（Gilbert–Elliott loss，GEL）。本节针对 CipherSpectrum 测试子集的 4121 条会话样本分别构造上述两种扰动，并使两种扰动造成的整体丢包率均保持约 10%。

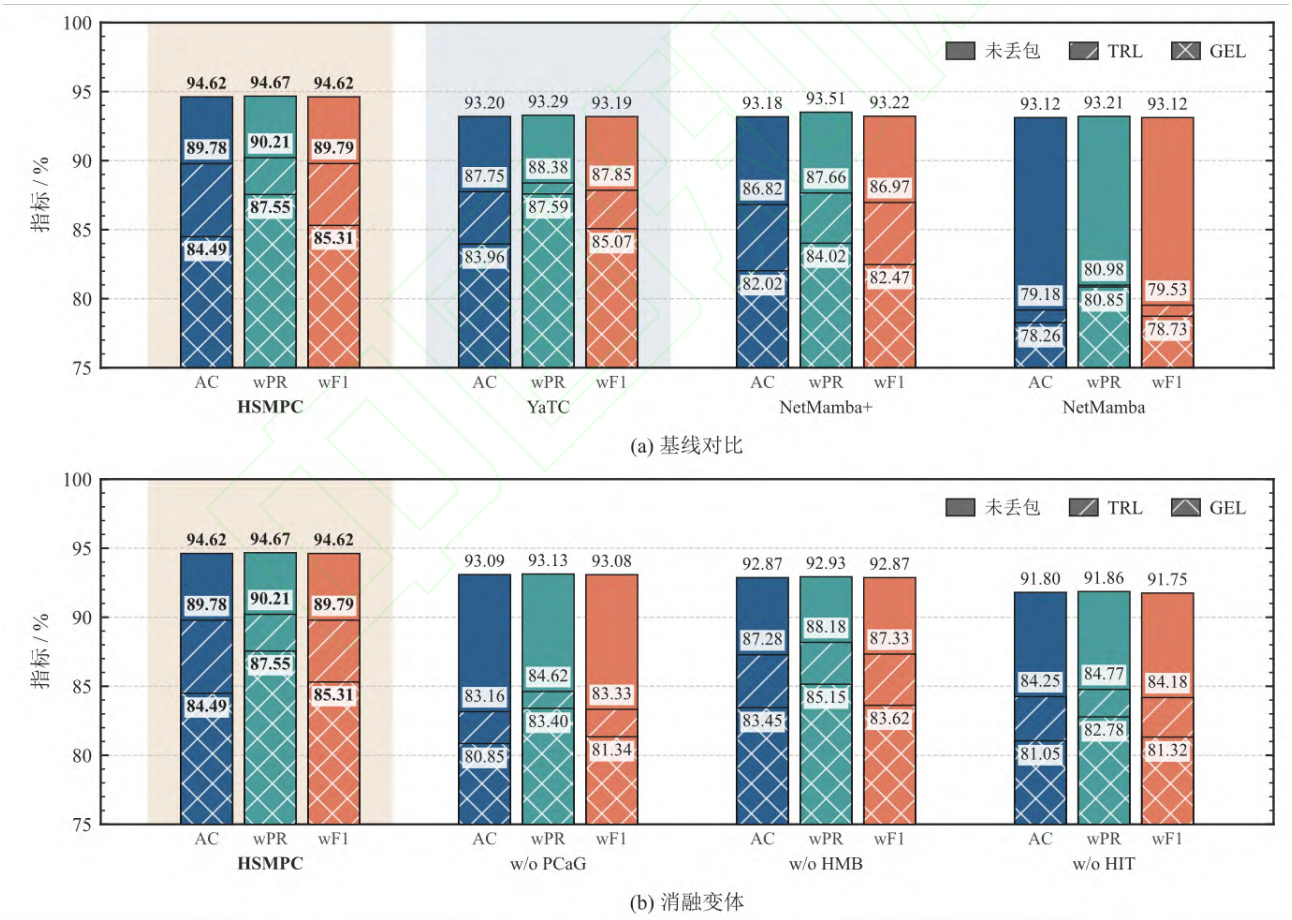


图 5 HSMPC 及其消融变体与轻量化基线在扰动情况下的分类性能

Fig.5 Classification performance of HSMPC and its ablation variants compared with lightweight baselines under disturbance

图 5 给出 YaTC、NetMamba、NetMamba+与 HSMPC 及其消融变体在完整流量与施加不同类型扰动后流量的分类结果。各模型均使用完整捕获流量进行微调得到的检查点，并针对丢包流量进行推

理，测试过程中不进行再训练或参数更新。因此，该实验主要反映模型迁移至不完整流量观测时的跨条件泛化能力

HSMPC 在流量捕获完整时整体指标较优，但施加扰动后的性能降幅揭示了更细致的趋势。在四种非消融方法中，YaTC 在 GEL 扰动下的三项指标降幅均最小，而 HSMPC 在 TRL 扰动下的三项指标降幅均最小。HSMPC 在 GEL 扰动下的绝对性能优势很大程度上得益于其在完整流量上更高的性能起点，YaTC 对原始分类性能的保持能力反而更强。在 TRL 扰动下，HSMPC 同时具备更高的性能起点和更小的性能降幅，因而其领先优势进一步扩大。NetMamba+在当前两类扰动下的退化幅度较大，可能与其同时依赖字节、包大小和时序模态更易受到丢包状态影响有关。

不同消融变体上的扰动结果则表明，各模块的贡献在当前扰动条件下仍保持正向。HMB 在 TRL 与 GEL 条件下相对 w/o HMB 的性能差异与完整流量条件处于相近量级，说明在当前约 10%的扰动设置下，结构调制未表现出相对于该对照更大的额外退化。HIT 与 PCaG 的增益在扰动下明显扩大，其中 w/o PCaG 变体在两类扰动下降幅最大，可能反映独立参数化的两级编码器对输入分布偏移更为敏感。

4　结　论

为充分利用层级结构信息、实现跨粒度交互建模，本文提出基于层级结构调制与参数耦合的加密流量分类方法 HSMPC。

1）在 CSTNET-TLS1.3、CipherSpectrum 和 LFETT2021 数据集上，3 个随机种子的平均加权 F1 分别达到 92.59%、94.62%和 97.36%，领先多数轻量化方法，表现出较好的分类性能与稳定性。

2）模型仅包含 2.17×10^6 的预训练参数和 1.91×10^6 的平均微调参数，在 A800 GPU 上的单样本模型前向延迟约为 3.5 ms，显存占用 22.70 MB，实现了分类精度、推理延迟与显存开销之间的良好平衡。

3）消融实验表明，HIT、HMB 和 PCaG 均能有效提升分类性能。层级结构信息与结构调制使模型形成结构条件化的状态保持与更新机制，而参数耦合则使字节序列调用与流级包序列调用在同一参数空间中累积梯度贡献，联合提升建模能力。

未来工作将针对固定裁剪与填充造成的无效字节预算及有效信息感知不足问题，研究内容密度驱动的动态自适应输入机制，根据样本信息密度自动筛选和分配字节预算，减少冗余填充与关键内容截断，进一步提升模型的分类性能与推理效率。

参考文献（References）

[1] 陈子涵，程光，徐子恒，等．互联网加密流量检测、分类与识别研究综述[J]．计算机学报，2023，46（5）：1060-1085．
Chen Z H,Cheng G,Xu Z H,et al.A survey on internet encrypted traffic detection,classification and identification[J].Chinese Journal of Computers,2023,46(5):1060-1085(in Chinese).

[2] 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(in Chinese).

[3] Gu A,Goel K,Ré C.Efficiently modeling long sequences with structured state spaces[C/OL]//International Conference on Learning Representations.2022[2026-07-15].https://openreview.net/forum?id=uYLFoz1vlAC.

[4] Gu A,Dao T.Mamba:linear-time sequence modeling with selective state spaces[C/OL]//First Conference on Language Modeling.2024[2026-07-15].https://openreview.net/forum?id=tEYskw1VY2.

[5] Taylor V F,Spolaor R,Conti M,et al.Robust smartphone app identification via encrypted network traffic analysis[J].IEEE Transactions on Information Forensics and Security,2018,13(1):63-78.

[6] Barradas D,Santos N,Rodrigues L,et al.FlowLens:enabling efficient flow classification for ML-based network security applications[C]//Proceedings 2021 Network and Distributed System Security Symposium.Reston:Internet Society,2021.

[7] Liu C,He L T,Xiong G,et al.FS-Net:a flow sequence network for encrypted traffic classification[C]//IEEE INFOCOM 2019 - IEEE Conference on Computer Communications.Piscataway:IEEE,2019:1171-1179.

[8] Yao H P,Liu C,Zhang P Y,et al.Identification of encrypted traffic through attention mechanism based long short term memory[J].IEEE Transactions on Big Data,2022,8(1):241-252.

[9] Devlin J,Chang M W,Lee K,et al.BERT:pre-training of deep bidirectional Transformers for language understanding[C]//Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers).Stroudsburg:Association for Computational Linguistics,2019:4171-4186.

[10] Dosovitskiy A,Beyer L,Kolesnikov A,et al.An image is worth 16x16 words:Transformers for image recognition at scale[C/OL]//International Conference on Learning Representations.2021[2026-07-15].https://openreview.net/forum?id=YicbFdNTTy.

[11] He K M,Chen X L,Xie S N,et al.Masked autoencoders are scalable vision learners[C]//2022 IEEE/CVF Conference on Computer Vision and

Pattern Recognition (CVPR).Piscataway:IEEE,2022:15979-15988.
[12] Lin X J,Xiong G,Gou G P,et al.ET-BERT:a contextualized datagram representation with pre-training Transformers for encrypted traffic classification[C]//Proceedings of the ACM Web Conference 2022.New York:ACM,2022:633-642.
[13] Zhao R J,Zhan M W,Deng X W,et al.Yet another traffic classifier:a masked autoencoder based traffic Transformer with multi-level flow representation[C]//Proceedings of the AAAI Conference on Artificial Intelligence.Palo Alto:AAAI Press,2023,37:5420-5427.
[14] Zhou G M,Guo X W,Liu Z T,et al.TrafficFormer:an efficient pre-trained model for traffic data[C]//2025 IEEE Symposium on Security and Privacy (SP).Los Alamitos:IEEE Computer Society,2025:1844-1860.
[15] 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]//2025 IEEE 33rd International Conference on Network Protocols (ICNP).Piscataway:IEEE,2025:1-12.
[16] Chen X Y,Han L,Zhan D C,et al.MIETT:multi-instance encrypted traffic Transformer for encrypted traffic classification[C]//Proceedings of the AAAI Conference on Artificial Intelligence.Palo Alto:AAAI Press,2025,39:15922-15929.
[17] Yang L M,Liu L,Huang J J,et al.MM4flow:a pre-trained multi-modal model for versatile network traffic analysis[C]//Proceedings of the 2025 ACM SIGSAC Conference on Computer and Communications Security.New York:ACM,2025:1664-1678.
[18] Wang T Z,Xie X H,Wang W D,et al.NetMamba:efficient network traffic classification via pre-training unidirectional Mamba[C]//2024 IEEE 32nd International Conference on Network Protocols (ICNP).Piscataway:IEEE,2024:1-11.
[19] Wang T Z,Xie X H,Wang W D,et al.NetMamba+:a framework of pre-trained models for efficient and accurate network traffic classification[J].IEEE Transactions on Networking,2026,34:7361-7376.
[20] Zhao Y Q,Dettori G,Boffa M,et al.The sweet danger of sugar:debunking representation learning for encrypted traffic classification[C]//Proceedings of the ACM SIGCOMM 2025 Conference.New York:ACM,2025:296-310.
[21] RFC 6066 Transport Layer Security (TLS) extensions:extension definitions[S].
[22] RFC 8446 The Transport Layer Security (TLS) protocol version 1.3[S].
[23] Wickramasinghe N,Shaghaghi A,Tsudik G,et al.SoK:decoding the enigma of encrypted network traffic classifiers[C]//2025 IEEE Symposium on Security and Privacy (SP).Los Alamitos:IEEE Computer Society,2025:1825-1843.
[24] Zhang X X,Wei F R,Zhou M.HIBERT:document level pre-training of hierarchical bidirectional Transformers for document summarization[C]//Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics.Stroudsburg:Association for Computational Linguistics,2019:5059-5069.
[25] Stergiou A,Poppe R,Kalliatakis G.Refining activation downsampling with SoftPool[C]//2021 IEEE/CVF International Conference on Computer Vision (ICCV).Piscataway:IEEE,2021:10337-10346.
[26] Cho K,Mitsuya K,Kato A.Traffic data repository at the WIDE Project[C]//2000 USENIX Annual Technical Conference (USENIX ATC 00).Berkeley:USENIX Association,2000:263-270.
[27] Van Ede T,Bortolameotti R,Continella A,et al.FlowPrint:semi-supervised mobile-app fingerprinting on encrypted network traffic[C]//Proceedings 2020 Network and Distributed System Security Symposium.Reston:Internet Society,2020.
[28] Xu H B,Cheng Z Y,Li S H,et al.ProxyKiller:an anonymous proxy traffic attack model based on traffic behavior graphs[C]//Computer Security – ESORICS 2024.Cham:Springer Nature Switzerland,2024,14983:162-181.
[29] Gu Z Y,Gou G P,Hou C S,et al.LFETT2021:a large-scale fine-grained encrypted tunnel traffic dataset[C]//2021 IEEE 20th International Conference on Trust, Security and Privacy in Computing and Communications (TrustCom).Piscataway:IEEE,2021:240-249.
[30] Katharopoulos A,Vyas A,Pappas N,et al.Transformers are RNNs:fast autoregressive Transformers with linear attention[C]//Proceedings of the 37th International Conference on Machine Learning.Cambridge:PMLR,2020,119:5156-5165.
[31] Ali A A,Zimerman I,Wolf L.The hidden attention of Mamba models[C]//Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers).Kerrville:Association for Computational Linguistics,2025:1516-1534.