面向细粒度多网页浏览行为识别的报文级标注方法

顾玥[1,2]，陈力[3]，李丹[2,3]，高凯辉[3]
（1.清华大学深圳国际研究生院，广东 深圳 518055；2.清华大学计算机科学与技术系，北京 100084；
3.中关村实验室，北京 100194）

摘　要：针对多网页并发访问下混合加密流量中的细粒度多网页浏览行为识别难题，提出了一种基于报文间时序相关特征的报文级标注（PLL）方法。该方法通过结合一维卷积神经网络和多头注意力机制，学习同一网页中不同报文之间的局部和全局时序相关性特征，并采用一维转置卷积恢复特征至原始报文序列的长度，从而建立报文与其时序特征之间的精确对应关系。在此基础上，模型能够实现高精度的报文级标注，并进一步推断出用户浏览多个网页的时间信息，如各网页访问的起止时间与持续时间。实验结果表明，PLL 对访问开始时间和持续时间的识别准确率分别达到了 98% 和 97%，能够有效解决混合加密流量中多网页浏览行为识别的关键问题。

关键词：加密流量识别；多网页浏览行为识别；报文级标注
中图分类号：TP393
文献标志码：A
DOI：10.11959/j.issn.1000–436x.2025122

Packet-level labeling method for fine-grained multi-webpage browsing behavior recognition

GU Yue[1,2], CHEN Li[3], LI Dan[2,3], GAO Kaihui[3]
1. Tsinghua Shenzhen International Graduate School, Shenzhen 518055, China
2. Department of Computer Science and Technology, Tsinghua University, Beijing 100084, China
3. ZGC LAB, Beijing 100194, China

Abstract: To address the challenge of fine-grained multi-webpage browsing behavior recognition in mixed encrypted traffic under concurrent multi-webpage access, a packet-level labeling method based on inter-packet temporal correlation features, called PLL, was proposed. This method combined one-dimensional convolutional neural networks with multi-head attention mechanisms to learn both local and global temporal correlation features between different packets within the same webpage. Additionally, one-dimensional transposed convolution was employed to restore features to the original packet sequence length, thereby establishing a precise correspondence between each packet and its contextual features. This enabled accurate packet-level labeling and further supported the inference of user's browsing time information for multiple webpage, such as the start time, end time, and duration of each webpage visit. Experimental results show that PLL achieves 98% and 97% accuracy in identifying visit start time and duration, effectively solving the critical issue of multi-webpage browsing behavior recognition in mixed encrypted traffic.

Keywords: encrypted traffic recognition, multi-webpage browsing behavior recognition, packet-level labeling

收稿日期：2025–03–04；修回日期：2025–06–25
通信作者：李丹，tolidan@tsinghua.edu.cn
基金项目：国家重点研发计划基金资助项目（No.2022YFB3104900）；北京高校卓越青年科学家计划基金资助项目（No.JWZQ20240101008）
Foundation Items: The National Key Research and Development Program of China (No.2022YFB3104900), The Beijing Outstanding Young Scientist Program (No.JWZQ20240101008)

0　引言

当用户在浏览网站时，其细粒度的网页浏览行为蕴含了重要的隐私信息。例如，访问特定网页的时间和停留时间能够反映出用户的浏览习惯[1]、搜索意图[2]和个人偏好[3-4]，这些信息被恶意利用将导致严重的隐私泄露。因此，许多社交媒体平台实施了隐私保护机制，并广泛采用安全套接字层/传输层安全（SSL/TLS, secure sockets layer/transport layer security）等加密协议[5-6]，以确保数据传输的安全性。目前，超过80%的网站部署了SSL/TLS加密协议[3]。

尽管加密协议能够在一定程度上保护用户隐私安全，但已有大量网站和网页指纹识别研究[7-13]证明，从加密流量中识别出用户浏览的网站域名和网页URL是可行的。这些研究观察到，在浏览某个网站或网页过程中会产生独特的流量模式，从而能够区分不同网站或网页[10]。基于这一观察，这些方法通过分析网络流量模式，如数据报文的大小和到达时间间隔，能够准确识别出用户访问的网站域名或网页URL。这些研究揭示了现有隐私保护机制中易被流量分析利用的潜在安全漏洞，深化了对隐私泄露路径的理解，为未来隐私保护机制的设计与优化提供了重要的理论支撑与实践依据。

然而，现有的网站或网页指纹识别方法无法实现细粒度的多网页浏览行为识别，如各网页访问开始时间和结束时间。如图1中现有方法所示，由于用户在访问一个网站时常常同时浏览多个网页[14-15]，而HTTP/2多路复用机制[16]允许来自同一网站不同网页的流量通过同一TCP连接传输，因此报文与各网页的对应关系难以确定。这使提取每个网页的纯净流量变得非常困难，导致现有方法识别网页的准确率下降，并且无法确定网页访问开始时间和持续时间。

本文聚焦于以下研究问题：在多网页混合的加密流量中，攻击者是否不仅能够识别用户访问的具体网页，还能进一步推断其细粒度的多网页浏览行为，如各网页的访问起止时间与持续时间？

解决上述研究问题要求识别方法为每个报文分配相应的网页标签，如图1中报文级标注（PLL, packet-level labeling）方法所示。然而，由于报文内容是加密的，攻击者只能利用报文的大小、上下行方向等可用特征进行分析。因此，单个数据报文所包含的信息非常有限，这为实现精确的报文级标注带来了巨大的挑战。

为了解决上述问题，本文对网页流量进行了深入分析，并得出以下结论。1) 一个网页由多个文本、图片等网页元素组成，浏览器按照特定的时间顺序加载这些网页元素，这意味着同一网页不同元素对应的数据报文之间具有时序相关性。2) 客户端对特定元素的请求以及服务器对该请求的响应通常遵循固定的时间模式，这进一步表明了来自同一元素的数据报文之间也存在时序相关性。每个网页元素的加载顺序和各元素的请求-响应模式不受混合流量的干扰。此外，这些加载顺序和请求-响应模式在不同网页之间是不同的。因此，数据报文之间的时序相关性可以作为区分不同网页的有效特征，从而帮助实现精确的报文级标注。

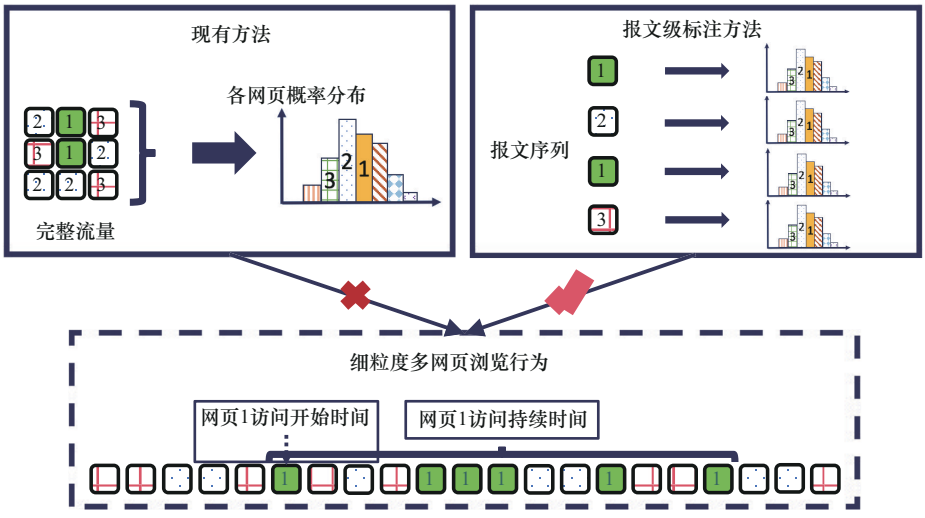


图1　现有方法和报文级标注方法对比

基于上述分析，本文提出了一种基于报文间时序相关特征的报文级标注方法。具体而言，PLL受语义分割算法的启发[17-19]，采用典型的编码器-解码器架构来识别混合流量中每个报文的网页类别。首先，PLL通过编码器捕捉报文序列中的上下文时序相关特征。编码器采用一维卷积神经网络（1DCNN, 1D convolutional neural network）提取局部上下文特征，学习同一网页元素的数据报文之间的时序相关性。在经过1DCNN处理后，编码器进一步通过多头注意力机制提取全局上下文特征，学习同一网页中不同元素数据报文之间的时序相关性。接下来，PLL利用解码器执行上采样，将编码器生成的压缩表征恢复至原始报文序列长度，建立报文与其时序相关特征之间的映射关系。随后，PLL对每个数据报文进行分类，实现精准的报文级标注。最后，通过分析每个网页对应的报文序列，PLL能够进一步推断用户细粒度的多网页浏览行为。

本文基于公开的真实网页流量数据集[3]，合成了一个大规模的多网页加密流量数据集，并在此数据集上对PLL进行了全面的评估。为了更真实地模拟多网页浏览行为，本文模拟了用户行为，包括每个网页的访问开始时间、结束时间和持续时间。此外，为了确保合成的数据集更贴近实际网络场景，本文还复现了HTTP/2多路复用机制和Web浏览器的加载策略。本文研究工作贡献如下。

1) 发现了现有方法在识别细粒度多网页浏览行为时存在一定的局限性。其主要原因在于实际浏览网页的过程中会生成多网页混合流量，其中不同网页的数据报文交织在一起，导致报文与网页之间缺乏明确的对应关系。

2) 提出了一种基于报文间时序相关特征的PLL方法，旨在实现细粒度的多网页浏览行为识别。PLL通过结合1DCNN和多头注意力机制，捕捉同一网页数据报文之间的局部和全局时序相关特征，从而实现精准的报文级标注。经过标注的报文序列可以用于进一步分析用户的多网页浏览行为。

3) 实现了PLL的原型，并在合成的大规模多网页加密流量数据集上进行了全面的评估。实验结果表明，PLL在报文级标注任务中表现出色，最高可以达到99%的召回率。在访问开始时间和持续时间的识别上，PLL分别可取得98%和97%的准确率，显著优于基线方法。此外，PLL在多网页识别任务中也表现卓越，实现了99%的真阳率，相较于基线方法提高了5%~79%。

1　相关工作

1.1　网站指纹识别

流量分析与识别一直是热点研究方向[20-23]。近年来，网站指纹识别因其在网络监管和用户行为分析等领域的潜在应用，已成为一个重要的研究课题[24-33]。当前的网站指纹识别方法通过分析网络流量的特征，如报文大小、上下行方向和到达时间等，来判断用户访问的网站域名。学术界已对网站指纹识别进行了广泛研究，为其理论基础和实际应用的发展做出了重要贡献。

1.1.1　单标签网站指纹识别

在单标签网站浏览场景中，攻击者假设用户每次上网仅访问一个网站，并且所有网络流量均与该网站相关联。这种场景较为简单，早期的网站指纹识别方法大多基于这一假设展开研究[13,34-35]。

在这些研究中，一些方法利用统计特征执行网站指纹识别。例如，CUMUL方法[13]使用104个统计特征并结合支持向量机（SVM, support vector machine）分类器，而Wang等[34]提出的方法则采用超过3 000个统计特征，并使用k-最近邻（kNN, k-nearest neighbor）分类器进行识别。另一些方法则更侧重于应用时序列特征进行分类。例如，深度指纹（DF, deep fingerprinting）方法[26]利用报文上下行方向序列，并使用卷积神经网络进行分类。在网站指纹识别领域中，这些不同的研究方法展示了特征选择与分类器设计的重要性。

1.1.2　多标签网站指纹识别

与单标签网站浏览场景相比，多标签网站浏览场景中的混合流量缺乏报文与网站间的清晰映射关系，使现有单标签网站指纹识别方法难以直接适用。Juarez等[14]的研究表明，当用户同时访问多个网站时，直接将现有的单标签网站指纹识别方法应用于多标签网站混合流量时，网站指纹识别的准确率会显著下降。这一挑战促使研究者们探索更为先进的技术，以应对复杂的多标签网站浏览场景。

1) 基于流量分割的方法

基于流量分割的方法通过利用统计特征、时间特征以及其他相关特征，从混合流量中分离出属于

单个网站的纯净流量，并识别对应的网站。Xu等[15]和Yin等[36]聚焦于同时访问2个网站的场景，并假设这2个网站的访问时间存在一定的时延。他们认为混合流量的初始部分主要是第一个网站的纯净流量，而当开始访问第二个网站时，会出现大量网络请求，具体表现为报文到达时间间隔的显著缩短与报文数量的急剧增加。为此，他们提出了BalanceCascade-XGBoost分割方法，用于准确识别第二个网站流量的起始点，并基于此从混合流量中分离出第一个网站的纯净流量。随后，他们使用随机森林和XGBoost算法对第一个网站进行识别。

Wang等[37]提出了一种假设，即在浏览同一网站时，连续报文之间的到达时间间隔存在一个上限。当报文之间的时间间隔超过此上限时，意味着用户开始访问第二个网站。基于这一假设，他们通过设置时间间隔上限来检测第二个网站流量的起始点，并从混合流量中分割出第一个网站的纯净流量。针对时间间隔上限设置不当可能造成的分割误差，他们分析了流量统计特征的变化，并结合kNN、LF-kNN和朴素贝叶斯（NB, naive Bayes）算法，进一步实现了更细粒度的流量分割。这种方法使他们能够更准确地从混合流量中提取第一个网站的纯净流量。随后，他们利用现有的单标签网站指纹识别方法对第一个网站进行识别。

Gu等[38]同样假设在访问2个网站时存在访问时延，并通过设置时延上限来分割第一个网站的纯净流量。为了进一步提高识别的准确性，他们利用细粒度特征，如往返时间（RTT, round-trip time）和第一个GET请求报文的大小，结合朴素贝叶斯算法对第一个网站进行识别。在成功识别第一个网站后，他们进一步去除混合流量中与第一个网站相关的粗粒度特征，如TCP连接数、上下行方向的报文数量和带宽，以便更准确地识别第二个网站。

Cui等[39]提出了一种假设，即混合流量的开始部分和结束部分由可分割的纯净流量组成，并且不同网站之间的重叠流量仅占混合流量的很小部分。基于这一假设，他们在混合流量的起始部分和结束部分设置了纯净流量长度阈值，从而能够有效地分割出纯净流量。在分割出纯净流量后，他们利用卷积神经网络和长短期记忆网络识别对应的2个网站。

此外，Cui等[9]进一步利用报文数量或持续时间对流量进行分段，并使用kNN算法对每段流量进行识别，通过多数投票机制确定混合流量中的网站标签。

2) 基于深度学习的方法

上述基于流量分割的多标签网站指纹识别方法主要关注纯净流量，忽略了多个网站之间的重叠流量。然而，重叠流量实际上包含了对网站识别起着重要作用的特征。为了充分挖掘混合流量中重叠部分的潜在特征，近年来的研究开始采用深度学习方法，从整个混合流量中提取特征，以更全面地利用这些重叠部分的潜在特征。这种方法的优势在于，它能够捕捉到流量中的复杂模式和潜在关联，从而显著提高网站识别的准确性。

Guan等[11]在已知同时打开网站数量的前提下，提出了块注意力分析模型（BAPM, block attention profiling model）架构，该架构利用完整的多网站混合流量（包括重叠流量）作为模型输入，以避免信息丢失。他们引入了一种基于报文方向序列的多页面感知表征方法，通过分组划分尽可能将不同网站的流量特征块分离出来，从而减少不同网站流量之间的干扰。接着，BAPM采用多头注意力机制将同一网站的特征块进行关联，这使模型能够在混合流量中有效识别出多个网站。

Deng等[10]提出了一种从完整的多网站混合流量中直接提取各网站特征的方法，并通过多标签分类识别流量中包含的多个网站。首先，该方法将流量分成多个区段，然后利用1DCNN提取每个区段的特征。接着，他们使用多头注意力机制计算各区段特征之间的相关性，以捕捉不同区段之间的潜在相关特征。最后，他们将这些相关特征连接起来，并通过多层感知机进行多标签分类，最终识别出流量中包含的多个网站。

Xu等[40]将多网站混合流量中的网站指纹识别问题转化为计算机视觉中的目标检测问题，提出了一种基于流量图的目标检测模型（TIOD, trace image-based object detection model）。具体来说，他们将多网页流量重叠问题建模为计算机视觉中的对象重叠问题，并引入S-矩阵策略将流量转换为图像。在此基础上，设计了一个名为网站指纹区域卷积神经网络（WF R-CNN, website fingerprinting region-based convolutional neural network）的目标检测模型，该模型能够从流量图中提取特征，进而在混合

流量中识别出潜在的网站。

1.2　网页指纹识别

相较于网站指纹识别主要关注用户访问的网站域名，网页指纹识别则专注识别用户在同一网站内访问的具体页面。这一技术的意义在于，它能够帮助网络监管人员进行细粒度的用户行为分析，深入了解用户在特定网站上的访问内容，并及时发现异常访问行为[3]。网页指纹识别相较于网站指纹识别的难点在于，同一网站下的网页使用了相似的网页模板和元素布局，导致其流量相似度较高，直接利用网站指纹识别方法进行同一网站下的网页识别准确率较低[3-4]。为了解决这一难题，现有的网页指纹识别方法考虑到下行方向报文序列中体现网页元素的内容信息，而不同网页之间主要是通过网页元素的内容进行区分。因此，一些方法选择使用从下行方向报文序列中提取的流量突发序列特征[3,41]。在此基础上，这些方法结合了传统机器学习模型，如随机森林、kNN 和决策树等，或者采用深度学习模型，如 1DCNN 等，以实现高效的网页识别。

现有的网页指纹识别方法通常假设用户在浏览某个网站时只打开一个网页。然而，实际情况是用户通常同时打开多个网页，这会产生多网页的混合流量。同样地，在这种情况下，报文与网页之间的映射关系缺失，导致难以准确地获取每个网页的纯净流量。因此，针对单一网页提出的网页指纹识别方法在多网页场景下往往会失效。尽管已有研究[42]尝试识别同一网站下的多个网页，且现有多标签网站指纹识别方法在一定程度上可迁移用于网页指纹识别，但这类方法普遍缺乏对同一网站下不同网页报文序列的有效分离能力，难以还原报文与网页之间的精确映射关系，从而无法进一步追踪用户细粒度的多网页浏览行为。因此，本文旨在提出一种报文级标注方法，为每个报文分配相应的网页标签，从而实现同一网站下不同网页报文序列的有效分离，进而推断用户细粒度的多网页浏览行为信息。

1.3　语义分割

语义分割是计算机视觉领域中的一项关键任务，其目标是充分利用图像中每个像素的上下文信息及其空间特征，将每个像素精确分配到特定的目标物体类别中，从而实现对图像中不同物体的精确识别和定位。这种细粒度的分割能力不仅能够区分图像中的各个目标，还能精确地描绘其边界和结构，因此在多个领域发挥着重要作用。例如，在自动驾驶领域中，语义分割被用于道路、车辆、行人和交通标志的精准检测。在医学影像分析领域中，它被应用于病灶区域的分割和诊断。而在遥感图像处理领域，则被用于地物分类、变化检测等任务。

目前，大多数语义分割模型，如全卷积网络（FCN, fully convolutional network）[17]、DeepLab 系列[43]、Mask R-CNN[44] 和 U-Net[19] 均采用编码器–解码器架构来高效学习和融合上下文信息。在这一架构中，编码器部分通过一系列的卷积层和池化操作，逐步降低图像的分辨率，同时提取层次化的特征表示。此外，编码器部分能够提取图像中的关键模式、纹理和结构特征，并有效捕获上下文信息，为后续分割提供强大的语义支持。

解码器部分则通过逐步恢复图像的分辨率，完成对特定类别区域的细化分割。解码器通常利用上采样（如反卷积或双线性插值）和额外的卷积操作，将编码器提取的高级语义特征与浅层细节特征结合，进一步优化分割结果。通过跳跃连接或特征融合的方式，解码器能够同时兼顾高分辨率的细节与低分辨率的全局语义信息，从而生成精确且具有完整边界的语义分割结果，输出最终的每像素类别概率图。上述方法的逐步发展和优化，显著推动了语义分割在学术研究和实际应用中的性能提升。

考虑到在报文级标注任务中，受限于加密传输，单个报文可用的信息通常仅包括大小、时间等基本特征，信息量十分有限。为提高报文级标注的准确性，需要充分结合上下文信息，挖掘隐藏的时序关系和全局特征。语义分割算法以其擅长整合局部与全局信息的特点，为该问题的解决提供了启发。因此，可以借鉴语义分割算法的核心思想，设计一种适用于报文级标注任务的模型框架，通过对上下文信息的有效整合与利用，提升报文级标注的精确度。

尽管图像领域的语义分割算法取得了显著进展，但直接将这些方法应用于网络流量数据时却面临诸多挑战。网络流量数据的本质是时间序列数据，缺乏图像数据所具有的空间连续性和直观的几何结构。这一差异导致传统依赖空间特征的图像分割模型难以在网络流量数据中直接发挥作用。因此，为了使这些图像分割模型适应网络流量数据的特性，亟须对模型进行针对性的调整和优化。

2　威胁模型

本节将详细描述研究的问题场景。如图 2 所示，Web 浏览器使用 SSL/TLS 加密协议保护用户的网络活动。当用户同时打开同一网站下的多个网页时，攻击者可以捕获用户在浏览过程中产生的多网页混合加密流量，试图从混合加密流量中追踪用户的细粒度多网页浏览行为，包括各网页的访问开始时间、结束时间以及持续时间等信息。这些信息可以被攻击者用来推测用户的上网习惯和偏好。潜在的攻击者包括入侵用户个人计算机的黑客、用户的互联网服务提供商（ISP, Internet service provider），或是伺机在路由器上嗅探流量的不法分子。

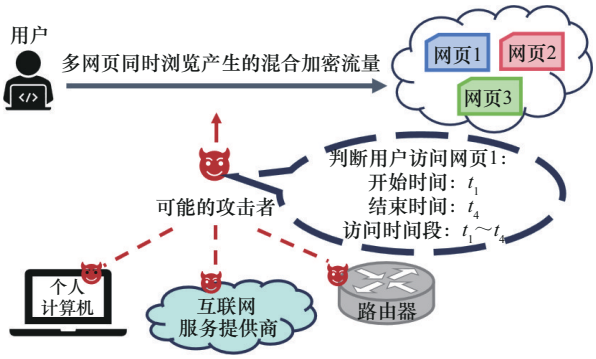


图 2　面向混合加密流量的细粒度多网页浏览行为识别威胁模型

与已有研究[10-11,14]保持一致，本文在封闭世界和开放世界场景下评估所提模型的性能表现。攻击者提前规定一个敏感网页集合。在封闭世界场景中，攻击者假设用户仅访问这些敏感网页，并且攻击者能够收集所有敏感网页的流量数据。在这种情况下，攻击者的目标是准确识别用户正在访问的具体敏感网页。相比之下，在开放世界场景中，攻击者假设用户可以访问任意网页，包括敏感网页集合外的非敏感网页。由于网页数量众多，攻击者无法收集所有网页的流量数据，因此在这种情况下，攻击者的目标是判断用户访问的是哪个敏感网页，还是非敏感网页。

3　方法设计

本节将详细介绍基于报文间时序相关特征的 PLL 方法的设计。首先，本节将概述 PLL 的整体架构，阐明其核心设计思想和功能目标。随后，将对 PLL 的各个组成部分进行深入探讨，以揭示其在实现精准报文标注中的关键机制与具体实现方式。

3.1　PLL 方法架构设计

PLL 旨在为每个数据报文分配相应的网页标签，以获取用户细粒度的多网页浏览行为信息。PLL 通过学习同一网页对应的数据报文之间的时序相关特征，实现准确的报文级标注。图 3 展示了基于报文间时序相关特征的 PLL 方法的架构。PLL 的输入为双向报文大小序列，其中正值表示上行方向的报文大小，负值表示下行方向的报文大小。选择双向报文大小序列作为输入的原因在于，不同网页所包含的网页元素差异会导致网页加载过程中产生的网络流量显著不同。因此，不同网页之间的报文大小序列有明显区别。此外，由于各网页的元素加载顺序及请求-响应时间的差异，不同网页对应的报文上下行方向序列也会有所变化。

受语义分割算法的启发，PLL 采用典型的编码器-解码器架构作为基本框架。首先，PLL 利用编码器提取时序上下文信息，将编码器中用于提取空间特征的操作替换为能够高效处理时间序列数据的模块 1DCNN 和多头注意力机制，以学习同一网页

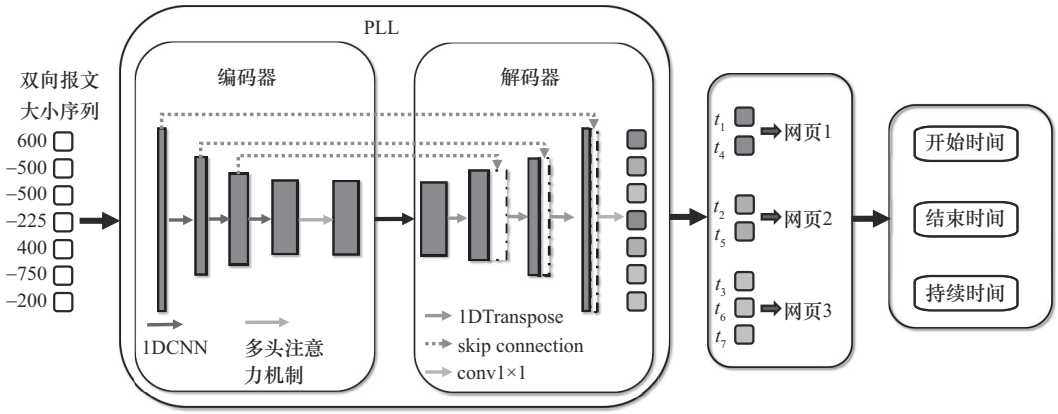


图 3　基于报文间时序相关特征的 PLL 方法架构

对应的报文之间的时序相关特征。接着，PLL通过解码器进行上采样，采用转置卷积层将编码器生成的压缩特征恢复到原始报文序列长度，建立报文与其时序相关特征之间的映射关系，随后计算每个报文属于各网页的概率分布。最后，PLL通过设置分类阈值，确定每个报文所属的网页，实现准确的报文级标注。

综上所述，通过分析每个网页对应的报文序列，PLL能够推断出用户细粒度的多网页浏览行为。基于PLL的报文级标注结果，可以识别每个网页的第一个报文和最后一个报文的时间戳，从而进一步推导出用户的多网页浏览行为，如每个网页的访问开始时间、结束时间和持续时间。

3.2　编码器设计

PLL通过编码器学习两类时序相关特征：一类是同一网页元素对应的报文之间的时序相关性，这些特征在流量中表现为局部上下文时序特征；另一类是同一网页的不同元素对应的报文之间的时序相关性，这些特征则在流量中表现为全局上下文时序特征。

1) 捕获局部时序特征。编码器首先使用1DCNN进行局部时序特征提取。由于1DCNN能够有效提取时序数据中的局部模式，因此非常适合捕获网页流量数据的局部时序特征[10]。为了进一步增强特征学习能力，本节在局部时序特征提取过程中引入了以下3种结构。

① 残差结构。残差结构是ResNet[45]等深度学习模型的关键组成部分。该结构允许一层的输出绕过一个或多个后续层，并与跳过层的输出相加，从而有效缓解梯度消失问题，促进更深层网络的训练。这一机制进一步增强了编码器在局部时序特征提取过程中的有效性和稳定性。

② 扩展卷积。通过在卷积核中引入间隙，在不增加计算复杂度的情况下有效地扩展了感受野[46]。这种方式有助于捕捉更全面的局部上下文时序特征相关信息，从而实现更加准确的特征表示。

③ 最大池化层。通过在指定的窗口内选择最大值来减小输入数据的空间维度。这一操作有助于减少计算开销，并有效防止过拟合[47]。通过保留最显著的特征，最大池化层使特征对小幅扭曲和变化具有更强的不变性，从而增强了模型的鲁棒性。

2) 捕获全局时序特征。在1DCNN之后，编码器利用多头注意力机制捕获不同网页元素对应的报文之间的长距离时序相关特征。多头注意力机制[48]基于传统的注意力机制[49]，通过并行应用多个注意力头，使模型能够同时关注报文序列中不同位置的特征信息。这种并行处理能力使模型能够有效捕捉来自不同网页元素报文之间的复杂长距离相关性，从而实现更准确的报文级标注。

3.3　解码器设计

PLL通过编码器处理后获得了压缩的低分辨率时序特征图。为了对每个报文进行标注，PLL通过解码器利用上采样将这些低分辨率时序特征图恢复到原始报文序列的长度，同时结合编码器早期阶段提取的细粒度时序特征，以确保解码器能够充分利用编码器提取的细节信息，实现精确的报文级标注。

1) 重建输出。解码器利用一维转置卷积[50]进行上采样，有效地逆转了常规卷积操作，从而扩展了压缩特征的时间维度，并建立了报文与其时序相关特征之间的映射关系。此外，解码器采用与编码器相同的残差结构和扩展卷积结构，进一步增强了重建时序相关特征的能力。

2) 保留细粒度信息。在上采样过程中，解码器引入了来自编码器的跳跃连接，使其能够结合编码器早期阶段生成的高分辨率时序特征图，这些时序特征图包含关于原始流量的细粒度时序特征。通过将这些高分辨率时序特征图与上采样特征图进行融合，解码器不仅能够保留局部和全局时序特征的细节信息，还能充分利用上采样特征图提供的更广泛上下文信息增强对时序相关特征的重建能力。

3) 获得报文级标注结果。经过最后一层转置卷积处理后，特征图中的每一行对应于每个报文的时序相关特征。解码器对特征图中的每一行执行1×1卷积操作[19]，基于学习到的报文时序相关特征生成网页类别分数，从而得到每个报文属于不同网页的概率分布。通过设置分类阈值，PLL能够确定每个报文的网页类别，从而实现精确的报文级标注。

算法1　基于报文间时序相关特征的PLL方法

初始化　滑动窗口大小w，编码器Encoder和解码器Decoder参数

输入　流量序列$\boldsymbol{S}$

输出　各报文对应的标签序列$\boldsymbol{Y}$

1) 循环

2)　for i=1 to length($\boldsymbol{S}$) do

3) 从流量序列中提取滑动窗口 $\boldsymbol{W\_i}=\boldsymbol{S}[i-w+1:i]$

4) 编码器生成包含时序关联关系的隐藏状态 $\boldsymbol{h\_i}$=Encoder($\boldsymbol{W\_i}$)

5) 解码器对隐藏状态进行解码，并计算置信度 $\boldsymbol{c\_i}$=Softmax(Decoder($\boldsymbol{h\_i}$))

6) 从置信度中选择概率最大的标签作为当前报文的标注，y_i=argmax($\boldsymbol{c\_i}$)

7) 将每个报文的标注结果加入结果列表中，$\boldsymbol{Y}$.append(y_i)

8) end for

3.4 PLL方法训练过程

在训练过程中，PLL使用报文级标签作为真实标签，并结合Dice损失（Dice Loss）[51]和交叉熵损失（CE Loss, cross-entropy loss）[52]来优化模型的报文级标注性能。如式(1)所示，Dice Loss用于衡量每个报文的预测标签与实际标签之间的重叠程度，特别注重小类别的正确分类，因此在处理类别不平衡的数据集时表现尤为有效。CE Loss则度量每个报文的预测概率分布与真实概率分布之间的差异，如式(2)所示，能够提供更加稳定的梯度，从而有助于优化分类任务。通过结合Dice Loss和CE Loss，如式(3)所示，PLL充分发挥了两者的优势，Dice Loss通过强调预测标签和实际标签的重叠程度提升模型报文级标注性能，而CE Loss则增强了报文分类的准确性，从而能够在报文级标注任务中实现更精确的标注结果。

Dice损失定义为

$$\text{Dice Loss} = 1 - \frac{2\sum_{i=1}^{N} p_i g_i}{\sum_{i=1}^{N} p_i + \sum_{i=1}^{N} g_i} \tag{1}$$

其中，p_i是第i个数据报文的预测二进制标签，g_i是第i个报文的真实二进制标签，N是报文总数。

CE损失定义为

$$\text{CE Loss} = -\sum_{i=1}^{N} g_i \ln p_i \tag{2}$$

其中，p_i是第i类的预测概率，g_i是第i类的真实二进制标签。

组合损失（Combined Loss）定义为

$$\text{Combined Loss} = \alpha\text{Dice Loss} + \beta\text{CE Loss} \tag{3}$$

其中，α和β是控制每个损失函数对组合损失函数贡献的加权因子。经实验证明，当α和β各取0.5时，训练出的模型报文级标注性能最佳。

3.5 基于PLL标注结果的多网页浏览行为识别

本节可以利用报文级标注的结果来识别细粒度的多网页浏览行为。具体而言，每个网页的第一个报文的时间戳标志着用户首次访问该网页的时刻，作为用户开始与网页交互的起点。类似地，最后一个报文的时间戳则表示网页访问的结束，标志着用户完成交互或网页加载的时刻。第一个报文与最后一个报文之间的时间跨度揭示了用户在该网页上花费的时间，从而为分析用户对该网页的感兴趣程度提供了有力依据。

这些信息为深入了解用户的浏览习惯和个人偏好提供了宝贵的线索，如用户偏好在何时浏览特定内容。这些洞察不仅可以广泛应用于用户行为分析、精准内容推荐等领域，还可以为个性化营销策略的制定提供有力支持。通过深入挖掘这些数据，企业和平台能够更加准确地把握用户需求，从而提升服务质量和用户体验。

4 实验

本节将PLL与当前最先进的方法进行比较，评估其在封闭世界和开放世界场景下的性能表现。所有实验均在多网页加密流量数据集上进行，重点考察报文级标注、多网页浏览行为识别和网页识别任务性能。

4.1 实验设置

4.1.1 实验环境

本节在一台Linux服务器上进行实验，该服务器配备了2个Intel®Xeon®CPU E5-2630 v3（共56核）、128 GB RAM内存、2 TB磁盘和4块NVIDIA H100 GPU。整个模型架构通过TensorFlow构建和训练，采用了TensorFlow的GPU版本，以充分利用服务器上的H100 GPU进行并行计算和模型加速。为了简化模型设计和实验过程，本节使用了TensorFlow提供的高级接口Keras。

4.1.2 数据集

本文方法依赖于具备报文级细粒度标签的多网页加密流量数据集进行训练和测试。然而，当前公开数据集中普遍缺乏此类标签信息，且在实际网络环境中获得报文级标签存在较大难度。因此，本节基于公开的真实单网页加密流量数据集[3]，通过合

成方式构建了带有报文级细粒度标签的多网页加密流量数据集。为了更接近真实世界场景的流量特征，本节首先将原始网页流量重构为HTTP请求-响应消息序列。随后，通过设置每个网页的访问开始时间、结束时间和持续时间，模拟用户的浏览行为。接着，根据时间戳对多个网页的消息序列进行排序，以确保消息顺序能够反映实际的网页访问过程。最后，本节实现了HTTP/2、TCP和IP协议的机制，以执行消息分帧、帧分段以及段打包等过程，最终生成了与多个网页对应的加密流量报文序列。

本节提出的数据集包含报文级细粒度标签，可用于报文级标注模型的训练。该数据集涵盖了3个网站在封闭世界和开放世界场景下的多网页浏览实例。对于每个网站，指定了30个敏感网页，同时在开放世界场景中额外指定了10 000个非敏感网页。在本文所提数据集中，每条流量数据包含的网页数量为2~5个。在封闭世界和开放世界场景中，每个网站分别包含10 000条训练数据和2 500条测试数据。其中，封闭世界场景下的每条数据仅包含敏感网页的流量，开放世界场景下的每条数据不仅包含敏感网页的流量，还包含非敏感网页的流量。

4.1.3　PLL模型参数设置

在本文实验中，PLL超参数设置如表1所示。

表1　　PLL超参数设置

超参数名称	取值
学习率	0.001
批量大小	64
优化器	Adam
L2正则化系数	0.01
卷积层数	5
转置卷积层数	5
注意力头数	2

以上超参数设置是在多次实验中经过调整与验证后得到的最佳配置，旨在确保模型能够在多网页加密流量数据集上实现最佳性能。

4.1.4　报文级标注与多网页浏览行为识别任务的基线模型

本节选择了Sectioning算法[9]和BalanceCascade-XGBoost分割方法[15]作为对比方法，以评估PLL在报文级标注和多网页浏览行为识别任务上的性能。选择这2种方法的原因在于，它们能够处理任意网页数量的混合流量。由于这2种方法并非专门为报文级标注任务设计，因此本节在保持其原有算法思想和模型参数的基础上，进行了适当的修改，以使其能够支持报文级标注任务。

对于Sectioning算法，首先根据预定义的段大小对混合流量进行分段，然后识别每个分段对应的网页。随后对每个报文根据其所在分段的网页类别进行标记，从而实现流量的报文级标注。

BalanceCascade-XGBoost分割方法最初是为识别2个网站混合流量中第二个网站的起始报文而提出的。本节对该方法进行了扩展，以检测不同网页流量之间的切换点，从而实现混合流量中不同网页流量的分离。随后，识别连续2个切换点之间的流量段对应的网页，并将每个报文标记为所属流量段的网页类别。

4.1.5　网页识别任务的基线模型

除了进行报文级标注和用户浏览行为识别任务的对比实验外，本节还评估了PLL在混合流量中进行网页识别的性能表现，并与现有方法进行了比较。本节选择了2种单一网页指纹识别方法WPF[3]和FineWP[4]，以及一种多标签网站指纹识别方法ARES[10]，作为对比方法。同样，在对比实验中，使用原论文里的模型参数。

为了使单一网页指纹识别方法适应多网页浏览场景，本节对这些单一网页指纹识别模型进行了修改，使其能够产生多标签输出。

4.1.6　报文级标注任务评价指标

本节使用3种广泛应用的语义分割指标来评估报文级标注的性能，分别为Dice系数（Dice）、精度（precision）和召回率（recall）。这些指标的计算式和原理如下。

1) Dice系数。Dice系数用于衡量预测标签与实际标签之间相似性的指标。它通过比较预测标签和真实标签之间的重叠程度来评估报文级标注的准确性。Dice系数越高，表示报文级标注准确性越好。其计算式为

$$\mathrm{Dice}=\frac{2|A\cap B|}{|A|+|B|} \tag{4}$$

其中，A表示实际标签集合，B表示预测标签集合。

2) 精度。精度用于评估在预测为正类别的报文中，有多少预测正确。精度越高，表示错分为正类别的报文越少。

$$\text{precision} = \frac{\text{TP}}{\text{TP} + \text{FP}} \tag{5}$$

其中，TP为模型正确预测为正类别的报文数，FP为误判为正类别的负类别报文数。

3) 召回率。召回率用于衡量模型正确标记属于特定网页的所有报文的能力。召回率越高，表示漏报的报文越少。

$$\text{recall} = \frac{\text{TP}}{\text{TP} + \text{FN}} \tag{6}$$

其中，FN表示被误判为负类别的正类别报文数。

4.1.7 多网页浏览行为识别任务评价指标

为了验证PLL对细粒度的多网页浏览行为识别的能力，本节使用以下4个指标来评估识别每个网页的访问开始时间、结束时间和持续时间的性能表现。

1) 访问时间准确率（VTA, visit time accuracy）。访问时间定义为用户访问某一网页的首个报文出现的时间，即该网页加载过程中第一个报文到达的时刻。为了评估PLL识别网页访问开始时间的准确性，本节将VTA定义为识别每个网页第一个报文的准确率，并允许一个小的误差范围（如δ ms）。如果预测报文的到达时间与实际第一个报文的到达时间差值在δ ms以内，则认为该网页的访问时间预测是正确的。VTA的计算式为

$$\text{VTA} = \frac{1}{N}\sum_{i=1}^{N}\left[I\left(\left|t_i^{(\text{first, pred})} - t_i^{(\text{first, true})}\right| \leqslant \delta\,\text{ms}\right)\right] \tag{7}$$

其中，N表示用户浏览的网页总数，$t_i^{(\text{first, pred})}$表示预测的网页$i$第一个报文的到达时间，$t_i^{(\text{first, true})}$表示实际的网页$i$第一个报文的到达时间。$I$表示指示函数，如果预测和实际报文到达时间的差绝对值小于或等于δ ms，则返回1，否则返回0。

2) 结束时间准确率（ETA, end time accuracy）。本节将网页的访问结束时间定义为该网页对应的最后一个报文的到达时间。为了评估识别访问结束时间的准确性，本节定义了ETA指标，该指标衡量的是正确预测每个网页最后一个报文到达时间的能力。计算式为

$$\text{ETA} = \frac{1}{N}\sum_{i=1}^{N}\left[I\left(\left|t_i^{(\text{last, pred})} - t_i^{(\text{last, true})}\right| \leqslant \delta\,\text{ms}\right)\right] \tag{8}$$

其中，$t_i^{(\text{last, pred})}$是预测的网页$i$最后一个报文的到达时间，$t_i^{(\text{last, true})}$是实际的网页$i$最后一个报文的到达时间。$I$是指示函数，如果预测的最后一个报文的到达时间与实际的最后一个报文的到达时间相差δ ms，则返回1，否则返回0。

3) 访问持续时间准确率（VDA, visit duration accuracy）。用于衡量同时正确预测网页第一个报文和最后一个报文的能力。如果预测的第一个报文和最后一个报文的到达时间与实际到达时间的误差都在δ ms范围内，则认为预测是正确的。VDA的计算式为

$$\text{VDA} = \frac{1}{N}\sum_{i=1}^{N}\left(I\left(\left|t_i^{(\text{first, pred})} - t_i^{(\text{first, true})}\right| \leqslant \delta\,\text{ms}\right)\&\left(\left|t_i^{(\text{last, pred})} - t_i^{(\text{last, true})}\right| \leqslant \delta\,\text{ms}\right)\right) \tag{9}$$

其中，I是指示函数，如果预测的第一个报文和最后一个报文的到达时间与实际到达时间的误差都在δ ms范围内，则返回1，否则返回0。

4) 交并比（IOU, intersection over union）。用于衡量预测的网页访问持续时间区间与实际访问持续时间区间的重叠程度。IOU定义为预测访问持续时间区间与实际访问持续时间区间的交集与它们的并集之比。IOU的计算式为

$$\text{IOU} = \frac{\left|\left[t^{(\text{first, pred})}, t^{(\text{last, pred})}\right] \cap \left[t^{(\text{first, true})}, t^{(\text{last, true})}\right]\right|}{\left|\left[t^{(\text{first, pred})}, t^{(\text{last, pred})}\right] \cup \left[t^{(\text{first, true})}, t^{(\text{last, true})}\right]\right|} \tag{10}$$

其中，$\left[t^{(\text{first, pred})}, t^{(\text{last, pred})}\right] \cap \left[t^{(\text{first, true})}, t^{(\text{last, true})}\right]$表示预测和实际访问持续时间区间的重叠部分，$\left[t^{(\text{first, pred})}, t^{(\text{last, pred})}\right] \cup \left[t^{(\text{first, true})}, t^{(\text{last, true})}\right]$表示预测和实际访问持续时间区间的并集。IOU取值范围从0到1，分数越高表示预测和实际访问持续时间区间的重叠程度越高，1表示完全重叠，0表示没有重叠。

经实验证明，当δ取值超过1 ms后，PLL及其对比方法的VTA、ETA和VDA指标趋于稳定，这说明多网页浏览行为识别误差大部分在1 ms以内。4.3节展示了将δ设置为1 ms的实验结果。

4.1.8 网页识别任务评价指标

为了评估在用户多网页浏览场景下的网页识别性能，本节使用如下2种现有网页指纹识别方法常

用的评价指标。

1) 真阳率（TPR, true positive rate）。TPR 表示在用户实际访问的所有网页中被正确识别的网页所占的比例。它实际上就是前面提到的召回率指标。

2) 假阳率（FPR, false positive rate）。FPR 是指未访问的网页中被错误地识别为已访问的网页所占的比例。FPR 越低，表示识别性能越好。

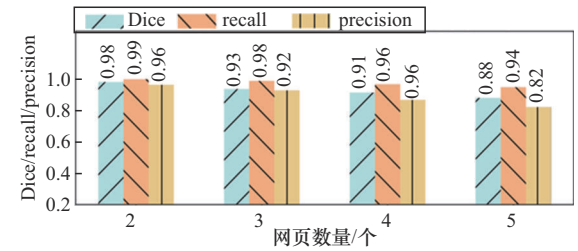
$$\mathrm{FPR}=\frac{\mathrm{FP}}{\mathrm{FP}+\mathrm{TN}} \tag{11}$$

其中，TN 表示正确识别为未访问的网页的数量。

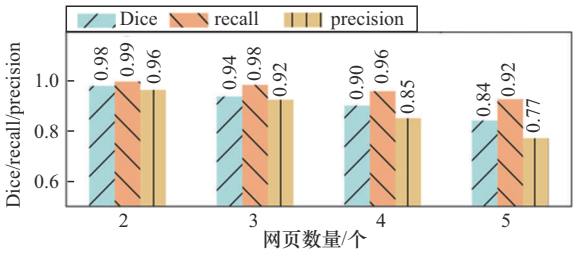
4.2　报文级标注实验评估结果

本节将 PLL 与 Sectioning 算法和 BalanceCascade-XGBoost 分割方法进行比较，以验证 PLL 报文级标注的能力。本节在包含不同网页数量（2~5 个网页）的多网页加密流量数据集上训练和测试模型，使用 Dice、recall 和 precision 指标评估模型的报文级标注性能。

图 4 为 PLL 的报文级标注评估结果。在封闭世界场景中，对于包含 2 个网页的混合流量数据集，PLL 的 Dice 最高能够达到 98%，recall 可以达到 99%，precision 可以达到 96%。即使在包含 5 个网页的混合流量数据集中，PLL 仍然能够达到 88% 的 Dice、94% 的 recall 和 82% 的 precision。在开放世界场景中，PLL 的性能表现与在封闭世界场景中相似，进一步验证了其性能的稳定性。这些实验结果表明，PLL 通过 1DCNN 和多头注意力机制学习到的网页报文间时序相关特征在不同网页数量的混合流量中保持稳定，几乎不受混合流量的干扰。因此，PLL 能够在各种网页数量的混合流量中实现精准的报文级标注。



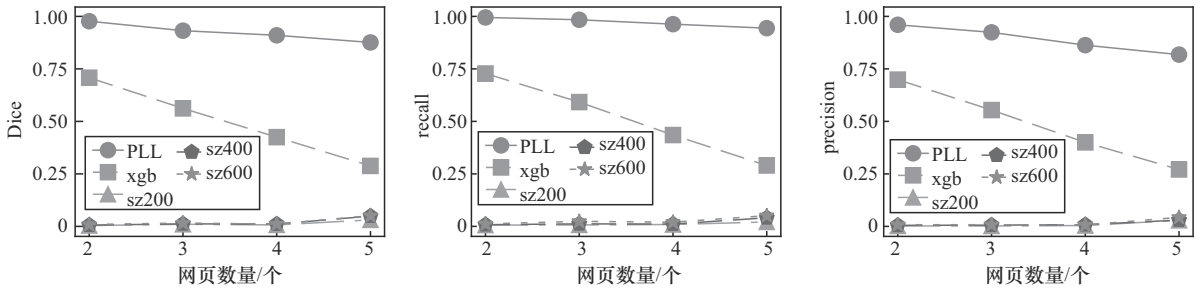
(a) 在封闭世界场景中不同网页数量下的评估结果



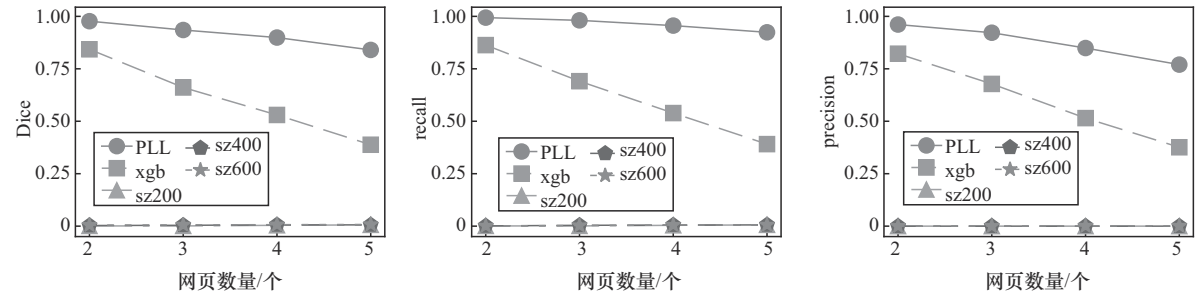
(b) 在开放世界场景中不同网页数量下的评估结果

图 4　PLL 的报文级标注评估结果

图 5 展示了 PLL 与基线方法的报文级标注实验对比结果。从图 5 中可以看出，PLL 在不同网页数量



(a) 在封闭世界场景中不同网页数量下的评估结果



(b) 在开放世界场景中不同网页数量下的评估结果

图 5　PLL 与基线方法的报文级标注实验对比结果

的混合流量数据集上的表现始终优于BalanceCascade-XGBoost分割方法（图5中标记为“xgb”）和Sectioning算法（图5中标记为“sz200”“sz400”和“sz600”）。在封闭世界场景中，BalanceCascade-XGBoost分割方法的Dice最高仅能达到73%，recall为74%，precision为71%。在开放世界场景中，BalanceCascade-XGBoost分割方法的Dice最高能达到84%，recall为86%，precision为79%。随着混合流量中网页数量的增加，BalanceCascade-XGBoost分割方法在封闭世界和开放世界场景中的报文级标注性能显著下降。在包含5个网页的混合流量数据集中，BalanceCascade-XGBoost分割方法的Dice甚至降至29%。

对于Sectioning算法，本节对不同的段大小（各分段报文数量取值为200、400和600）进行了对比实验。无论选择何种段大小，Sectioning算法的报文级标注性能始终较差。在封闭世界和开放世界场景下，Sectioning算法的最高precision分别低于5%和0.5%。这些实验结果表明，Sectioning算法和BalanceCascade-XGBoost分割方法在报文级标注任务中性能均表现不佳，未能学习到实现准确报文级标注所需的有效特征。

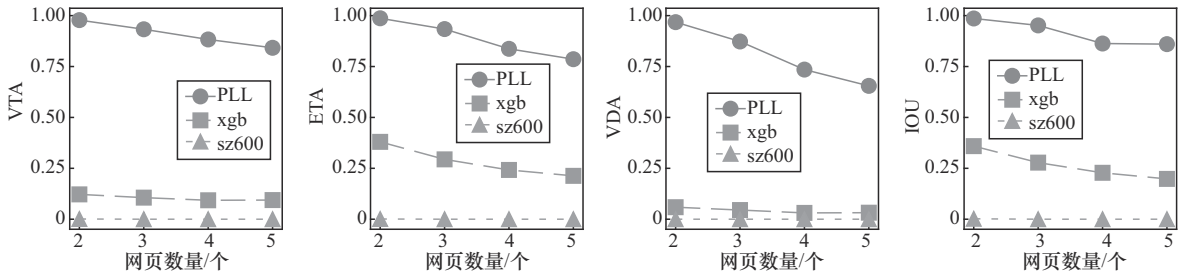
4.3　多网页浏览行为识别实验评估结果

本节评估了PLL、Sectioning算法（图6中标记为“sz600”）和BalanceCascade-XGBoost分割方法（图6中标记为“xgb”）利用各自的报文级标注结果推断多网页浏览行为的能力。具体而言，本节将上述3种方法直接应用于不同网页数量的混合流量数据集，以推断流量中每个报文所属的网页类别，从而获得各个网页对应的报文序列。随后，本节将计算以下4个评价指标：VTA、ETA、VDA和IOU，以对比3种方法的多网页浏览行为识别能力。

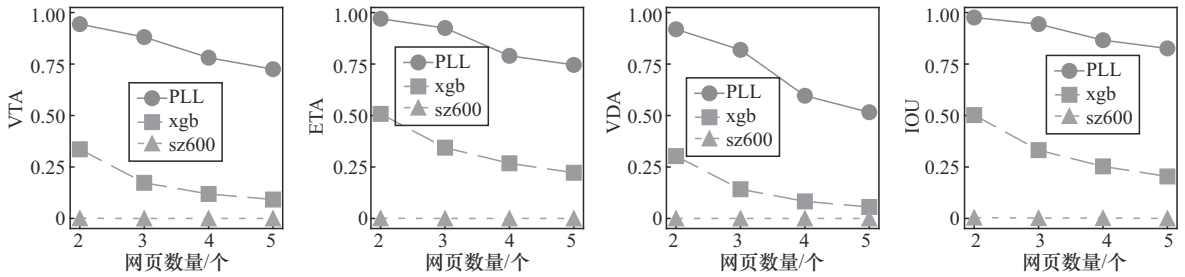
如图6所示，在识别每个网页的访问开始时间、结束时间和持续时间方面，PLL的性能表现明显优于其他方法。在2个网页的混合流量数据集中，PLL的VTA为98%，ETA为99%，VDA为97%，IOU为99%。随着网页数量的增加，PLL的性能略有下降。但即使在最差情况下，PLL仍能达到84%的VTA、79%的ETA、66%的VDA和86%的IOU，充分展示了PLL识别多网页浏览行为的能力。对比之下，BalanceCascade-XGBoost分割方法的各项指标均低于40%，在一些数据集上甚至低于10%。Sectioning算法则完全不具备多网页浏览行为识别的能力。这些实验结果说明了精准的报文级标注在多网页浏览行为识别任务中的重要性，同时再次凸显了报文间时序相关特征在实现精准报文级标注过程中的关键作用。

4.4　网页识别实验评估结果

为了验证PLL在多网页混合流量中的网页识别



(a) 在封闭世界场景中不同网页数量下的评估结果



(b) 在开放世界场景中不同网页数量下的评估结果

图6　多网页浏览行为识别实验结果

能力，本节将PLL与WPF、FineWP和ARES进行了比较，并使用网页识别任务的评价指标来评估模型的性能。

表2和表3给出了PLL和其他3个基线方法的网页识别实验结果。由表2可以看出，PLL可以达到99.8%的TPR，优于WPF和FineWP。这表明WPF和FineWP受多个网页混合流量的影响，而PLL由于具备学习报文间时序相关特征的能力，能够聚合来自同一网页的报文特征，从而能够准确识别多个网页。虽然ARES略逊于PLL，但在包含2、3、4个网页的混合流量数据集上也表现良好，仅在包含5个网页的混合流量数据集上表现较差。这表明同时打开的网页数量越多，ARES受到的性能影响越大。

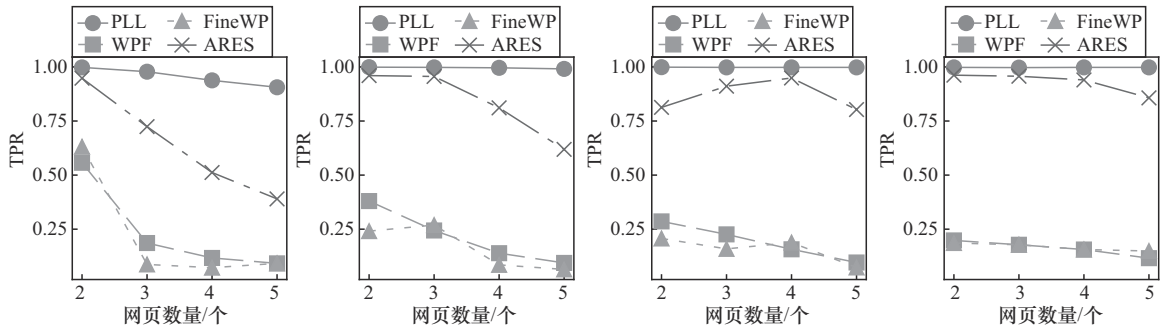
表2　网页识别实验结果(封闭世界场景)

方法	2个网页		3个网页		4个网页		5个网页	
	TPR	FPR	TPR	FPR	TPR	FPR	TPR	FPR
PLL	**0.998**	**0.004**	**0.998**	**0.005**	**0.998**	**0.005**	**0.998**	**0.009**
WPF	0.557	**0.004**	0.244	0.006	0.157	0.006	0.116	0.011
FineWP	0.631	0.006	0.269	0.006	0.188	0.008	0.149	0.011
ARES	0.948	0.005	0.956	0.006	0.949	0.006	0.857	0.010

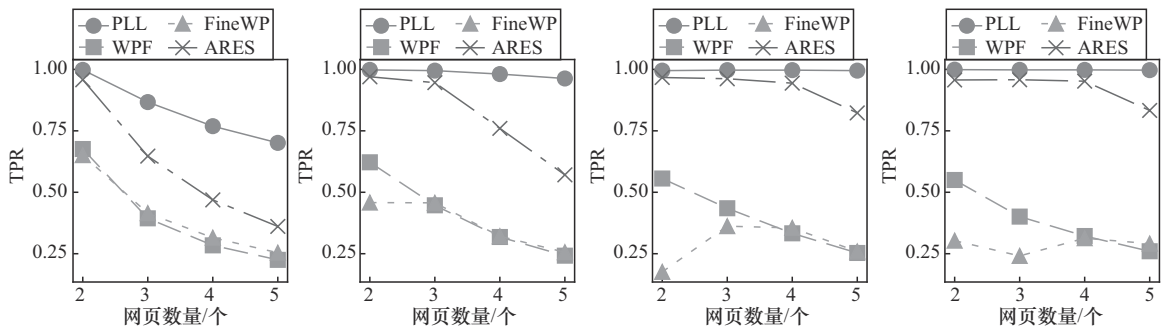
表3　网页识别实验结果(开放世界场景)

方法	2个网页		3个网页		4个网页		5个网页	
	TPR	FPR	TPR	FPR	TPR	FPR	TPR	FPR
PLL	**0.998**	**0.005**	**0.995**	0.007	**0.997**	**0.009**	**0.997**	**0.010**
WPF	0.676	0.006	0.447	**0.006**	0.333	0.010	0.260	**0.010**
FineWP	0.654	0.007	0.457	0.007	0.355	0.010	0.291	0.012
ARES	0.958	0.006	0.946	**0.006**	0.944	0.010	0.933	0.011

本节还进行了泛化性对比实验，比较了PLL与各基线方法在训练集和测试集网页数量完全不同的情况下的性能。这一实验旨在评估模型在面对全新、未见过的网页数量时的泛化能力，进一步验证PLL在处理具有高度变化的测试集时的鲁棒性和准确性。图7展示了网页识别泛化性对比实验中TPR指标的评价结果。可以看到，PLL在不同网页数量的混合流量数据集下，TPR变化较小，表现稳定。相比之下，WPF和FineWP的表现较差，其TPR甚至降至10%以下，显示出这些方法在处理不同网页数量时的较低准确性。ARES在不同网页数量下的性能波动较大，表明混合流量对ARES的干扰较为显著，可能影响了其在复杂数据集中的稳定性和准确性。



(a) 在封闭世界场景中不同网页数量下的评估结果



(b) 在开放世界场景中不同网页数量下的评估结果

图7　网页识别泛化性实验结果

5 局限性讨论

尽管本文方法在合成数据上展现出优异性能，其在真实网络流量数据集中的实际应用效果仍有待进一步验证。PLL方法对流量采集过程中报文级细粒度标签的准确性和精度提出了较高要求。然而，在复杂且大规模的真实网络环境中，获取高精度的报文级标签具有一定挑战性。一旦缺乏此类标签数据集，PLL方法的性能可能会受到限制。

未来研究可从以下2个方向展开探索：其一是提升在实际网络环境中采集并生成高精度报文级标签的能力，以增强模型的训练基础；其二是研究如何有效减轻模型对精细标签的依赖，如引入更高层次的序列建模、自监督学习或无监督特征学习策略，以提升模型在弱标签甚至无标签条件下的适应能力与整体识别能力。

6 结束语

本文提出了一种基于报文间时序相关特征的PLL方法，旨在实现细粒度的多网页浏览行为识别。PLL结合了1DCNN和多头注意机制，能够有效捕捉同一网页的数据报文之间的时序相关性特征，从而实现精准的报文级标注。基于对流量序列的报文级标注结果，可以进一步识别用户的多网页浏览行为。本文实现了PLL的原型，并在一个大规模的多网页加密流量数据集上进行了全面的实验评估。实验结果表明，PLL在报文级标注、多网页浏览行为识别和网页识别等任务中，均显著优于现有的基线方法，能够在混合流量环境下实现高精度的识别（VTA可达99%），展现出了良好的实际应用价值。本文方法可部署于用户终端或中间网络节点，始终仅基于加密流量的外部特征进行分析，并未涉及对用户实际数据内容的解密与获取，因此能够有效保障用户上网数据安全。此外，还进一步探讨了本文方法的局限性，并提出了未来的研究方向。

参考文献：

[1] LIANG T P, LAI H J. Discovering user interests from web browsing behavior: an application to Internet news services[C]//Proceedings of the 35th Annual Hawaii International Conference on System Sciences. Piscataway: IEEE Press, 2002: 2718-2727.

[2] CHENG Z C, GAO B, LIU T Y. Actively predicting diverse search intent from user browsing behaviors[C]//Proceedings of the 19th International Conference on World Wide Web. New York: ACM Press, 2010: 221-230.

[3] WANG K L, ZHANG J Z, BAI G D, et al. It's not just the site, it's the contents: intra-domain fingerprinting social media websites through CDN bursts[C]//Proceedings of the Web Conference 2021. New York: ACM Press, 2021: 2142-2153.

[4] SHEN M, LIU Y T, ZHU L H, et al. Fine-grained webpage fingerprinting using only packet length information of encrypted traffic[J]. IEEE Transactions on Information Forensics and Security, 2020, 16: 2046-2059.

[5] FREIER A, KARLTON P, KOCHER P. The secure sockets layer (SSL) protocol version 3.0[R]. 2011.

[6] RESCORLA E. The transport layer security (TLS) protocol version 1.3[R]. 2018.

[7] CAI X, ZHANG X C, JOSHI B, et al. Touching from a distance: website fingerprinting attacks and defenses[C]//Proceedings of the 2012 ACM Conference on Computer and Communications Security. New York: ACM Press, 2012: 605-616.

[8] CHEN S Y, CHEN S W, HE H S, et al. Causality correlation and context learning aided robust lightweight multi-tab website fingerprinting over encrypted tunnel[C]//Proceedings of the IEEE INFOCOM 2024-IEEE Conference on Computer Communications. Piscataway: IEEE Press, 2024: 761-770.

[9] CUI W Q, CHEN T, FIELDS C, et al. Revisiting assumptions for website fingerprinting attacks[C]//Proceedings of the 2019 ACM Asia Conference on Computer and Communications Security. New York: ACM Press, 2019: 328-339.

[10] DENG X H, YIN Q L, LIU Z T, et al. Robust multi-tab website fingerprinting attacks in the wild[C]//Proceedings of the 2023 IEEE Symposium on Security and Privacy (SP). Piscataway: IEEE Press, 2023: 1005-1022.

[11] GUAN Z, XIONG G, GOU G P, et al. BAPM: block attention profiling model for multi-tab website fingerprinting attacks on tor[C]//Proceedings of the Annual Computer Security Applications Conference. New York: ACM Press, 2021: 248-259.

[12] RAHMAN M S, SIRINAM P, MATHEWS N, et al. Tik-Tok: the utility of packet timing in website fingerprinting attacks[J]. arXiv Preprint, arXiv: 1902.06421, 2019.

[13] PANCHENKO A, LANZE F, ZINNEN A, et al. Website fingerprinting at Internet scale[C]//Proceedings of the 23rd Annual Network and Distributed System Security Symposium. Internet Society, 2016.

[14] JUAREZ M, AFROZ S, ACAR G, et al. A critical evaluation of website fingerprinting attacks[C]//Proceedings of the 2014 ACM SIGSAC Conference on Computer and Communications Security. New York: ACM Press, 2014: 263-274.

[15] XU Y X, WANG T, LI Q, et al. A multi-tab website fingerprinting attack[C]//Proceedings of the 34th Annual Computer Security Applica-

tions Conference. New York: ACM Press, 2018: 327-341.

[16] BELSHE M, PEON R, THOMSON M. Hypertext transfer protocol version 2 (HTTP/2)[R]. 2015.

[17] SHELHAMER E, LONG J, DARRELL T. Fully convolutional networks for semantic segmentation[C]//Proceedings of the IEEE Transactions on Pattern Analysis and Machine Intelligence. Piscataway: IEEE Press, 2017: 640-651.

[18] BADRINARAYANAN V, KENDALL A, CIPOLLA R. SegNet: a deep convolutional encoder-decoder architecture for image segmentation[J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2017, 39(12): 2481-2495.

[19] RONNEBERGER O, FISCHER P, BROX T. U-Net: convolutional networks for biomedical image segmentation[C]//Medical Image Computing and Computer-Assisted Intervention-MICCAI 2015. Berlin: Springer, 2015: 234-241.

[20] DONG S, XIA Y J, PENG T. Network abnormal traffic detection model based on semi-supervised deep reinforcement learning[J]. IEEE Transactions on Network and Service Management, 2021, 18(4): 4197-4212.

[21] DONG S. Multi class SVM algorithm with active learning for network traffic classification[J]. Expert Systems with Applications, 2021, 176: 114885.

[22] XIA Y J, DONG S, PENG T, et al. Wireless network abnormal traffic detection method based on deep transfer reinforcement learning[C]//Proceedings of the 2021 17th International Conference on Mobility, Sensing and Networking (MSN). Piscataway: IEEE Press, 2021: 528-535.

[23] DONG S, XIA Y J, WANG T. Network abnormal traffic detection framework based on deep reinforcement learning[J]. IEEE Wireless Communications, 2024, 31(3): 185-193.

[24] HAYES J, DANEZIS G. K-fingerprinting: a robust scalable website fingerprinting technique[J]. arXiv Preprint, arXiv: 1509.00789, 2015.

[25] PANCHENKO A, NIESSEN L, ZINNEN A, et al. Website fingerprinting in onion routing based anonymization networks[C]//Proceedings of the 10th Annual ACM Workshop on Privacy in the Electronic Society. New York: ACM Press, 2011: 103-114.

[26] DANEZIS G, DIAZ C. A survey of anonymous communication channels[J]. 2008.

[27] 邹鸿程, 苏金树, 魏子令, 等. 网站指纹识别与防御研究综述[J]. 计算机学报, 2022, 45(10): 2243-2278.
ZOU H C, SU J S, WEI Z L, et al. A review of the research of website fingerprinting identification and defense[J]. Chinese Journal of Computers, 2022, 45(10): 2243-2278.

[28] 杨宏宇, 宋成瑜, 王朋, 等. 洋葱路由器网站指纹攻击与防御研究综述[J]. 电子与信息学报, 2024, 46(9): 3474-3489.
YANG H Y, SONG C Y, WANG P, et al. Website fingerprinting attacks and defenses on tor: a survey[J]. Journal of Electronics & Information Technology, 2024, 46(9): 3474-3489.

[29] TAN X B, PENG C, XIE P, et al. Inter-flow spatio-temporal correlation analysis based website fingerprinting using graph neural network[J]. IEEE Transactions on Information Forensics and Security, 2024, 19: 7619-7632.

[30] MITSEVA A, PANCHENKO A. Stop, don't click here anymore: boosting website fingerprinting by considering sets of subpages[C]//Proceedings of the 33rd USENIX Security Symposium (USENIX Security 24). Piscataway: IEEE Press, 2024: 4139-4156.

[31] XIE Y, FENG J H, HUANG W J, et al. Contrastive fingerprinting: a novel website fingerprinting attack over few-shot traces[C]//Proceedings of the ACM Web Conference 2024. New York: ACM Press, 2024: 1203-1214.

[32] MENG W W, MA C, DING M, et al. Beyond single tabs: a transformative few-shot approach to multi-tab website fingerprinting attacks[C]//Proceedings of the ACM on Web Conference 2025. New York: ACM Press, 2025: 1068-1077.

[33] ZOU H C, SU J S, WEI Z L, et al. Toward an effective few-shot website fingerprinting attack with quadruplet networks and deep local fingerprinting features[J]. IEEE Transactions on Dependable and Secure Computing, 2025, PP(99): 1-18.

[34] WANG T, CAI X, NITHYANAND R, et al. Effective attacks and provable defenses for website fingerprinting[C]//Proceedings of the 23rd USENIX Security Symposium (USENIX Security 14). Piscataway: IEEE Press, 2014: 143-157.

[35] SIRINAM P, IMANI M, JUAREZ M, et al. Deep fingerprinting: undermining website fingerprinting defenses with deep learning[C]//Proceedings of the 2018 ACM SIGSAC Conference on Computer and Communications Security. New York: ACM Press, 2018: 1928-1943.

[36] YIN Q L, LIU Z T, LI Q, et al. An automated multi-tab website fingerprinting attack[J]. IEEE Transactions on Dependable and Secure Computing, 2021, 19(6): 3656-3670.

[37] WANG T, GOLDBERG I. On realistically attacking tor with website fingerprinting[J]. Proceedings on Privacy Enhancing Technologies, 2016, 2016(4): 21-36.

[38] GU X D, YANG M, LUO J Z. A novel Website Fingerprinting attack against multi-tab browsing behavior[C]//Proceedings of the 2015 IEEE 19th International Conference on Computer Supported Cooperative Work in Design (CSCWD). Piscataway: IEEE Press, 2015: 234-239.

[39] CUI W Q, CHEN T, CHAN-TIN E. More realistic website fingerprinting using deep learning[C]//Proceedings of the 2020 IEEE 40th International Conference on Distributed Computing Systems (ICDCS). Piscataway: IEEE Press, 2020: 333-343.

[40] XU Y F, WANG L M, CHEN J, et al. Tor trace in images: a novel multi-tab website fingerprinting attack with object detection[J]. The Computer Journal, 2024, 67(8): 2690-2701.

[41] SHEN M, GAO Z B, ZHU L H, et al. Efficient fine-grained website fingerprinting via encrypted traffic analysis with deep learning[C]//Proceedings of the 2021 IEEE/ACM 29th International Symposium on

Quality of Service (IWQOS). Piscataway: IEEE Press, 2021: 1-10.

[42] ZHAO X Y, DENG X H, LI Q, et al. Towards fine-grained webpage fingerprinting at scale[C]//Proceedings of the 2024 on ACM SIGSAC Conference on Computer and Communications Security. New York: ACM Press, 2024: 423-436.

[43] CHEN L C, PAPANDREOU G, KOKKINOS I, et al. DeepLab: semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected CRFs[J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2018, 40(4): 834-848.

[44] HE K M, GKIOXARI G, DOLLÁR P, et al. Mask R-CNN[C]//Proceedings of the 2017 IEEE International Conference on Computer Vision (ICCV). Piscataway: IEEE Press, 2017: 2980-2988.

[45] TARG S, ALMEIDA D, LYMAN K. Resnet in resnet: generalizing residual architectures[J]. arXiv Preprint, arXiv: 1603.08029, 2016.

[46] YU F, KOLTUN V. Multi-scale context aggregation by dilated convolutions[J]. arXiv Preprint, arXiv: 1511.07122, 2015.

[47] MURRAY N, PERRONNIN F. Generalized max pooling[C]//Proceedings of the 2014 IEEE Conference on Computer Vision and Pattern Recognition. Piscataway: IEEE Press, 2014: 2473-2480.

[48] HAN K, XIAO A, WU E H, et al. Transformer in transformer[J]. Advances in Neural Information Processing Systems, 2021, 34: 15908-15919.

[49] VASWANI A, SHAZEER N, PARMAR N, et al. Attention is all you need[J]. Advances in Neural Information Processing Systems, 2017.

[50] GAO H Y, YUAN H, WANG Z Y, et al. Pixel transposed convolutional networks[J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2020, 42(5): 1218-1227.

[51] SOOMRO T A, AFIFI A J, GAO J B, et al. Strided U-Net model: retinal vessels segmentation using dice loss[C]//Proceedings of the 2018 Digital Image Computing: Techniques and Applications (DICTA). Piscataway: IEEE Press, 2018: 1-8.

[52] MAO A Q, MOHRI M, ZHONG Y T. Cross-entropy loss functions: theoretical analysis and applications[J]. arXiv Preprint, arXiv: 2304.07288, 2023.

[作者简介]

顾玥（1997– ），女，黑龙江绥化人，清华大学博士生，主要研究方向为加密流量识别、流量预测等。

陈力（1988– ），男，广东江门人，博士，中关村实验室副研究员、博士生导师，主要研究方向为网络空间安全、大模型应用、高性能网络等。

李丹（1981– ），男，四川阆中人，博士，清华大学教授、中关村实验室双聘专家、博士生导师，主要研究方向为数据中心网络、网络安全、网络智能等。

高凯辉（1996– ），男，江西南昌人，博士，中关村实验室助理研究员，主要研究方向为数字孪生网络、大模型系统、智能网络等。