基于对比学习和预训练Transformer的流量隐匿数据检测方法

何帅[1,2]，张京超[1,2]，徐笛[1,2]，江帅[1,2]，郭晓威[1,2]，付才[1,2]
（1.华中科技大学网络空间安全学院，湖北 武汉 430040；2.分布式系统安全湖北省重点实验室，湖北 武汉 430040）

摘　要：为解决海量加密流量难表征、恶意行为难感知以及隐私数据归属难识别的问题，提出了一种基于对比学习和预训练Transformer的流量隐匿数据检测方法。考虑加密流量的高度复杂性、非结构化的特点以及传统下游任务的微调方法在加密流量领域的效果不佳的挑战，数据报文通过提取数据包序列被转换为类似自然语言处理中的词元。然后利用预训练Transformer模型将浅层表征转换为适用于多种加密流量下游任务的通用流量表征。通过将流量中的隐匿数据检测问题转换为相似性分析问题，基于对比学习的思想设计了一种差异性敏感的Transformer模型架构，同时使用样本的正负样本对增强模型对流量差异性的感知能力，并提出使用信息对比估计作为加密流量下游任务微调的损失函数。实验结果表明，所提方法在检测准确率、精确率、召回率以及F1分数等方面优于主流方法。

关键词：流量隐匿数据检测；预训练Transformer模型；对比学习；加密流量
中图分类号：TP393
文献标志码：A
DOI：10.11959/j.issn.1000–436x.2025043

Traffic concealed data detection method based on contrastive learning and pre-trained Transformer

HE Shuai[1,2], ZHANG Jingchao[1,2], XU Di[1,2], JIANG Shuai[1,2], GUO Xiaowei[1,2], FU Cai[1,2]
1. School of Cyberspace Science and Engineering, Huazhong University of Science and Technology, Wuhan 430040, China
2. Hubei Key Laboratory of Distributed System Security, Wuhan 430040, China

Abstract: To solve the problems of characterizing representing massive encrypted traffic, perceiving malicious behaviors, and identifying the ownership of privacy data, a traffic concealed data detection method was proposed based on contrastive learning and pre-trained Transformer. Considering the high complexity, unstructured nature of encrypted traffic, and the insufficient performance of traditional fine-tuning methods for downstream tasks in the encrypted traffic domain, data packets were first transformed into tokens which was similar to those used in natural language processing. Then, a pre-trained Transformer model was utilized to convert shallow representations into a general traffic representation, which was suitable for various encrypted traffic downstream tasks. By transforming the problem of concealed data detection into a similarity analysis problem, a diversity-sensitive Transformer architecture was developed leveraging contrastive learning, which enhanced the model's sensitivity to traffic differences through the use of positive and negative sample pairs, and using information noise contrastive estimation (Info NCE) as the loss function for fine-tuning downstream tasks of encrypted traffic. Experimental results show that the proposed method outperforms mainstream methods in terms of accuracy, precision, recall and F1 score.

Keywords: traffic concealed data detection, pre-trained Transformer model, contrastive learning, encrypted traffic

收稿日期：2024–10–17；修回日期：2025–03–11
通信作者：付才，fucai@hust.edu.cn
基金项目：国家重点研发计划基金资助项目（No.2023YFB3106402）；国家自然科学基金资助项目（No.62072200）
Foundation Items：The National Key Research and Development Program of China (No.2023YFB3106402), The National Natural Science Foundation of China (No.62072200)

0 引言

网络通信流量随着移动通信技术的发展，近年来呈现几何级增长的趋势[1]。由于加密技术不断普及，越来越多的应用程序和网站开始采用超文本传输安全协议（HTTPS, hypertext transfer protocol secure）等加密协议来保护用户数据。这种加密趋势使网络流量中隐匿数据占比显著攀升，隐匿数据是指流量报文中隐藏的、不易被察觉的数据，其不仅包括潜在的恶意行为，也涵盖流量报文中的用户隐私数据，用户在移动网络中使用的应用类型，虚拟专用网络中的服务和应用类型等。这种现象极大地增加了网络流量监控与分析的复杂性。因此，精准检测流量报文中的隐匿数据，对于增强网络环境的可靠性与安全性具有至关重要的意义。

根据特征提取范式的不同，流量检测方法可以分为基于流量报文的检测方法和基于特征工程的检测方法。基于流量报文的检测方法[2-4]主要针对超文本传输协议（HTTP, hypertext transfer protocol）明文流量场景，通过构建包含URL路径、请求参数等显式要素的攻击签名库，对后续流量进行模式匹配以识别恶意请求。此类方法虽然具有检测规则明确和部署成本较低的优势，但其局限性在加密流量占比日益增加的现代网络环境中愈发显著。首先，加密载荷的不可解析性导致传统签名匹配机制完全失效。其次，基于预定义规则的检测范式难以有效应对零日攻击等未知威胁。为实现对加密流量的有效分析，研究人员提出了基于特征工程的检测方法[5-9]。通过设计握手协议特征、数据流统计特征等，结合支持向量机、随机森林等模型实现加密流量分析。这类方法虽然在一定程度上提升了加密流量的可检测性，但其特征工程过程高度依赖领域知识，且模型表现出明显的任务特异性。即针对不同的检测任务（如恶意流量识别、应用分类、用户行为分析等）需要重新设计特征并独立训练模型，这不仅制约了检测系统的泛化能力，也降低了部署的效率。

近年来，基于预训练模型[10-11]的检测方法为构建通用化流量分析提供了新思路，此类方法首先通过自监督学习从海量未标注的流量数据中提取深度语义表征，再通过少量标注数据微调即可适配不同的下游任务。文献[12]验证了预训练模型在加密流量移动应用识别任务和虚拟专用网络服务类型识别任务中的有效性，但其采用的常规微调策略在隐匿数据检测场景中面临双重挑战：一方面，流量中的隐匿数据具有多维异构特性，即包含用户隐私信息等语义稀疏数据，也涉及恶意行为特征等潜在威胁模式，传统微调策略难以建立跨模态语义关联；另一方面，隐匿数据检测需同时满足实体识别、归属判定和威胁评估的复合需求，而现有的微调框架缺乏跨层级特征交互机制，导致模型在下游任务的检测效果受限。因此，需要研究细粒度的表征学习与微调方法，实现海量报文中多元化隐匿数据的高效感知与精准识别。

针对以上挑战，本文提出了一种基于对比学习和预训练Transformer[11]的流量隐匿数据检测模型。具体来说，本文首先将流量报文转换为类似自然语言处理中的词元结构，然后利用预训练Transformer模型学习流量报文的通用表征。接下来，为了更好地感知流量样本间的差异性，本文基于对比学习的思想设计了一种同时包含正负样本对的预训练模型微调方法，并提出使用信息对比估计作为损失函数。实验结果表明，本文方法能够以83.29%的准确率（ACC, accuracy）实现数据报文中隐私数据的归属精准检测，对比现有检测方法显著提升了5.27%。此外，本文方法同样适用于加密流量的恶意应用检测、移动应用分类与虚拟专用网络流量分类任务，通过在公开数据集上与9个近期主流方法的对比实验，本文方法在恶意应用检测、移动应用分类和虚拟专用网络流量分类任务上，分别达到了99.39%、97.68%和99.68%的准确率，优于FS-Net[13]、PERT[12]、ET-BERT[14]等近期发表的同领域高质量工作。本文主要的贡献如下。

1) 提出了研究流量隐私数据归属检测问题，通过将分类问题转化为同源性分析问题，实现了准确识别流量载荷中的隐私数据归属。本文有利于促进研究更鲁棒的用户隐私数据保护方法，同时有利于流量信息泄露的溯源技术的发展。

2) 提出了一种基于对比学习和预训练Transformer的流量隐匿数据检测模型。通过将输入的正负样本对同时引入对比学习网络，模型能够学习流量样本间的差异性，形成了更高效的流量检测下游任务微调范式。

3) 通过在多个公开数据集上与9个主流方法的对比，验证了本文方法在流量隐私数据归属检测等

问题的准确率、精确率（PR, precision）、召回率（RC, recall）、F1 分数等指标均优于现有主流方法，同时具有良好的适应性，在加密恶意软件、虚拟专用网络以及移动应用分类任务上的准确率也优于对比的主流方法。

1　相关工作

本文研究的主要问题是移动应用流量包含隐私数据的归属检测问题，其中隐私数据在本文场景中是指包含用户个人信息（如手机号、身份证号等）的数据报文。与该问题最相关的研究领域是加密流量分类领域[15-17]。在本文方法中，首先使用预训练模型对流量进行表征，随后利用对比学习方法使同一人员的隐私流量在表征的分布上更趋于相近，不同人员的隐私流量在表征的分布上更具差异性。因此，本节从加密流量分类、基于预训练的流量表征方法和基于相似度的检测方法 3 个方面展开介绍。

1.1　加密流量分类

基于深度学习的方法在加密流量分类中展示出了巨大的应用潜力[1,18]，包括研究目标定位[19-20]、数据收集[21]、数据预处理[22]、模型选择[23-24]、训练和评估[25]、应用校验改进[26-27]等步骤，通常被称为加密流量分类的“六步法”[28]。在数据预处理阶段，需要对数据包进行过滤、填充、截断和归一化等操作[29]。特征提取是关键步骤，可以基于原始数据包、流量特征或它们的组合实现[19]，但目前学术界对流量特征的分类还没有完全统一。Sirinam 等[30]利用数据包方向序列与卷积神经网络（CNN, convolutional neural network）实现了对 Tor 流量网站的精准识别。文献[19]提出了将流量字节转换为灰度图，然后使用卷积神经网络进行虚拟专用网络流量的分类。Liu 等[13]使用递归神经网络提出了一种端到端的加密流量分类模型 FS-Net，深入挖掘流量的潜在序列特征。Zeng 等[31]提出了深度全范围流量检测框架，利用深度学习技术，不需要人工干预和设计特征工程，有效地提升了加密流量分类和入侵检测任务的准确率，同时大幅减少了存储资源需求。虽然基于深度学习的方法不需要手动设计特征且数据处理的速度较高，但此类方法需要大量有标签的数据进行专用模型的训练。

1.2　基于预训练的流量表征方法

Lin 等[14]提出了一种名为 ET-BERT 的基于预训练技术的加密流量分类方法。该方法利用大规模加密流量学习一系列用于加密流量分类任务的通用数据报表示。文献[14]还提出了 2 个流量特定的自监督预训练任务，用于捕获字节级和词元嵌入级的上下文关系，获得通用的数据报表示。这种方法可以应用到多个加密流量场景任务中，如通用加密应用分类、加密恶意软件分类、虚拟专用网络加密流量分类、Tor 上加密应用分类等 5 个加密流量分类任务上。但 ET-BERT 主要关注加密流量分类、恶意软件检测等工作，本文模型更关注通过识别不同应用流量之间的相似性，实现隐私数据的归属检测。

1.3　基于相似度的检测方法

这类方法首先从网络流量中提取关键信息，如包的大小、传输协议、时间间隔、IP 地址等，用来构建特征向量[32-34]，然后通过计算特征向量之间的相似度来判断流量的相似性[35]。何高峰等[36]提出了使用多个最近邻（NN, nearest neighbor）过滤器的基于相似性的特征提取方法，提取基于相似性的特征、静态特征、节点流特征和分段的组合特征，用于大规模稀疏流量预测。而特征通常由人工提取，高度依赖于专家的知识和经验，但网络流量具有复杂性和多样性，手工设计的特征具有相似度度量单一、噪声敏感、动态性差、忽略时序信息等局限性。此外，在训练数据有限的情况下，模型易出现过拟合问题。

综上所述，由于现有的模型和方法面临着缺乏可解释性、特征提取和模型效果受限于数据集的质量、手工设计的特征局限性较大等挑战，目前尚缺乏有效手段从移动应用分类任务中高效识别隐私流量的人员归属的方法。

2　模型架构

2.1　模型总述

本文的目标是通过学习大规模流量报文的关联关系，利用同源性分析的比对方法，识别流量中的隐匿数据。为高效实现这一目标，本文首先对原始流量进行数据格式的转换，使用预训练模型学习流量的通用表示，最后引入改进的对比学习进行相似度检测。本文的模型框架如图 1 所示。

1) 流量表征。这一阶段是将应用级流量进行多次处理，转化为模型易于识别和处理的嵌入表示。

2) 预训练 Transformer 模型。为了学习流量语

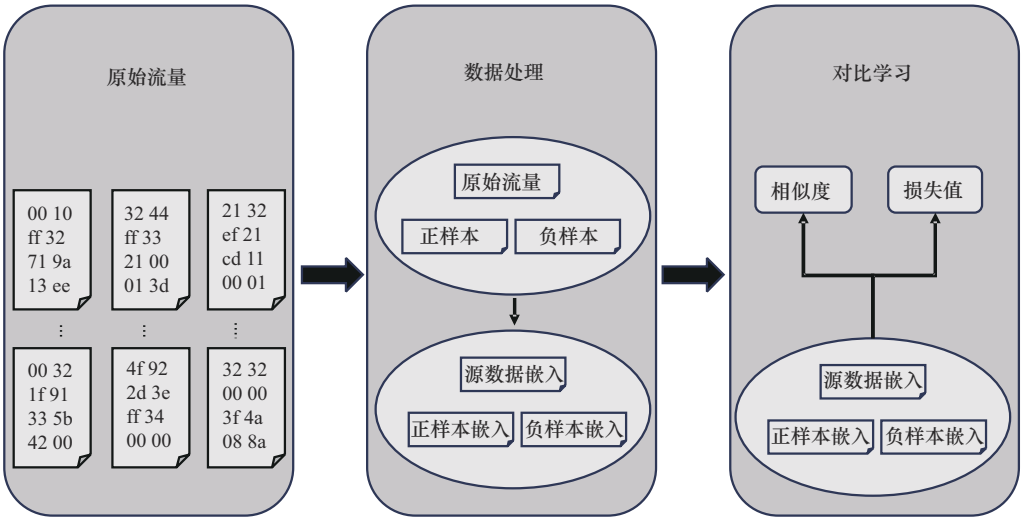

图 1　基于对比学习和预训练 Transformer 的流量隐匿数据检测模型框架

义的相似性，本文使用预训练 Transformer 模型学习具有大规模未标记数据的通用流量与其正负样本的区别，从而形成适用于多个下游任务的流量报文通用嵌入，具体在 2.3 节描述。

3) 基于对比学习的流量隐匿数据检测。为了实现对流量报文中的隐匿数据的准确检测，本文引入了对比学习的思想，通过对流量数据进行相似性分析，实现流量中的隐私数据的用户归属、恶意应用识别等。具体来说，本文提出将流量及其正负样本同时经过预训练模型得到三重表征，并在对应的下游任务上使用多层感知机进行微调，输出是流量报文包含隐匿数据的相似度得分。如果输出的相似度得分越高，则认为它们包含相似隐匿数据的概率越大，具体在 2.4 节描述。

2.2　流量表征

由于庞大的流量对数据的组织形式各有不同，使用原始流量直接作为网络的输入很难高效地学习到流量的通用表示、与正负样本的相似性和差异性。因此，本文采用将数据报文转换为词元（Datagram2Token）的方式，对网络流量进行处理和转化，形成类似自然语言处理中的词元（token），供模型学习。Datagram2Token 由以下 3 个步骤组成。

1) 提取数据包序列。数据包序列指的是一组源到目标（客户端到服务器或服务器到客户端）的相邻数据包，通常来自同一会话流。通过数据包序列的提取，能够捕获应用层的传输模式。本文在对数据进行处理时，首先将会话流中的连续数据包提取为数据包序列，以便描述网络流量的传输模式。

2) 数据包序列转化为词元。该步骤将数据包序列提取的加密数据报文转换为词元序列，以便可以应用于模型的预训练。为了将数据包序列表示转换为可用于预训练的词元表示，本文首先使用二元语法（bi-gram）模型编码十六进制的数据包序列。二元语法模型将十六进制的数据分割成连续的双字节（即每 2 个十六进制数为一组 bi-gram）序列。例如，对于十六进制数据 1A2B3C4D，分割成 1A2B 和 3C4D，将分割后的双字节序列化为一个序列，每个元素代表一个双字节对。然后采用字节对编码（BPE, byte-pair encoding）方法，进一步处理 bi-gram 模型生成的序列。BPE 首先初始化一个字符级别的词汇表，每个字符都是一个独立的词元，然后计算所有双字节对在数据中出现的次数，选择数据中出现频率最高的双字节对，将其合并后更新词汇表。经过对上述步骤的不断重复，直到达到预定的词汇表大小或合并次数，最终将 bi-gram 模型编码后的数据包序列转换为词元表示。每个词元的范围是从 0~65 535，表示可能的字节对组合。字典大小$|\mathcal{V}|$最多可以表示 65 536 种不同的词元。为了辅助模型预训练任务，在转换过程中还引入了几个特殊的词元，包含：① 分类标志词元，用于表示完整序列的代表词元；② 分隔标志词元，用于分隔不同段落；③ 填充词元，用于填充词元以满足最小长度要求；④ 遮盖词元，在预训练时用于遮蔽特定的词元以学习上下文。

3) 词元转换为嵌入（Token2Embedding）。在该步骤中，将词元、位置和段信息组合，使每个词

元转化为 768 维的向量，用于预训练模型的输入嵌入。词元嵌入是从前一步骤中生成的词元表示（通过查找表学习到的词元向量），维度设置为 768。位置嵌入确保模型通过相对位置学习到流量中词元之间的时间关系。段嵌入用于区分子段，以便在预训练中学习报文内部的上下文关系。将这些嵌入组合起来，能够构成一个完整的词元嵌入，作为 Transformer 模型的输入。每个词元嵌入都会经过 Transformer 网络多层编码，最终形成用于下游任务的流量表示。

经过以上 3 个步骤，网络流量样本可转化为用于对比学习和下游任务的多元融合嵌入。图 2 概述了如何通过流量表征优化网络流量数据的分类和分析，最终用于下游任务的过程。

2.3　预训练模型

在大规模流量报文中，字节之间没有明显的语义关联，因此需要使用自监督学习的预训练模型捕获字节之间的依赖关系。具体来说，本文使用加密流量掩码预训练模型[14]，其利用 Transformer 模型的双向特性，同时考虑了前向和后向的信息，在预训练阶段，输入的数据包序列中的每个词元都有 15% 的概率被随机掩码。被选中的词元有 80% 的概率被替换为特殊符号 MASK，有 10% 的概率被替换为一个随机词元，另外 10% 的概率保持不变。在这种随机掩码机制下，模型能够学习到词元嵌入之间的上下文关系，从而通过训练可以达到根据上下文预测被屏蔽位置的内容这一效果。

加密流量掩码预训练模型也被用于损失函数的计算，损失值是基于负对数的似然估计。模型通过预训练能够预测掩码位置的词元，捕捉上下文中词元之间的依赖关系，通过最大化被掩码词元的预测概率，根据损失函数所得的损失值来指导模型调整参数。损失值的计算式为

$$\mathrm{Loss} = -\sum_{i=1}^{k} \log\left(p\left(\mathrm{MASK}_i = \mathrm{token}_i \mid \bar{X}; \theta\right)\right) \quad (1)$$

其中，θ 表示模型参数集合，$\bar{X}$ 是 X 的掩码表示，MASK_i 表示序列中第 i_{th} 位置的掩码，概率 p 由 Transformer 编码器用 θ 来建模，表示模型根据上下文 $\bar{X}$ 预测 MASK_i 处的词元 token_i 的概率。

2.4　对比学习架构

为了增强模型感知流量差异性的能力，本文引入了对比学习的思想，因其典型的网络结构，孪生神经网络近几年在代码相似性检测任务[37]上表现优异[32]。因此，本文提出了使用相似性对比的思

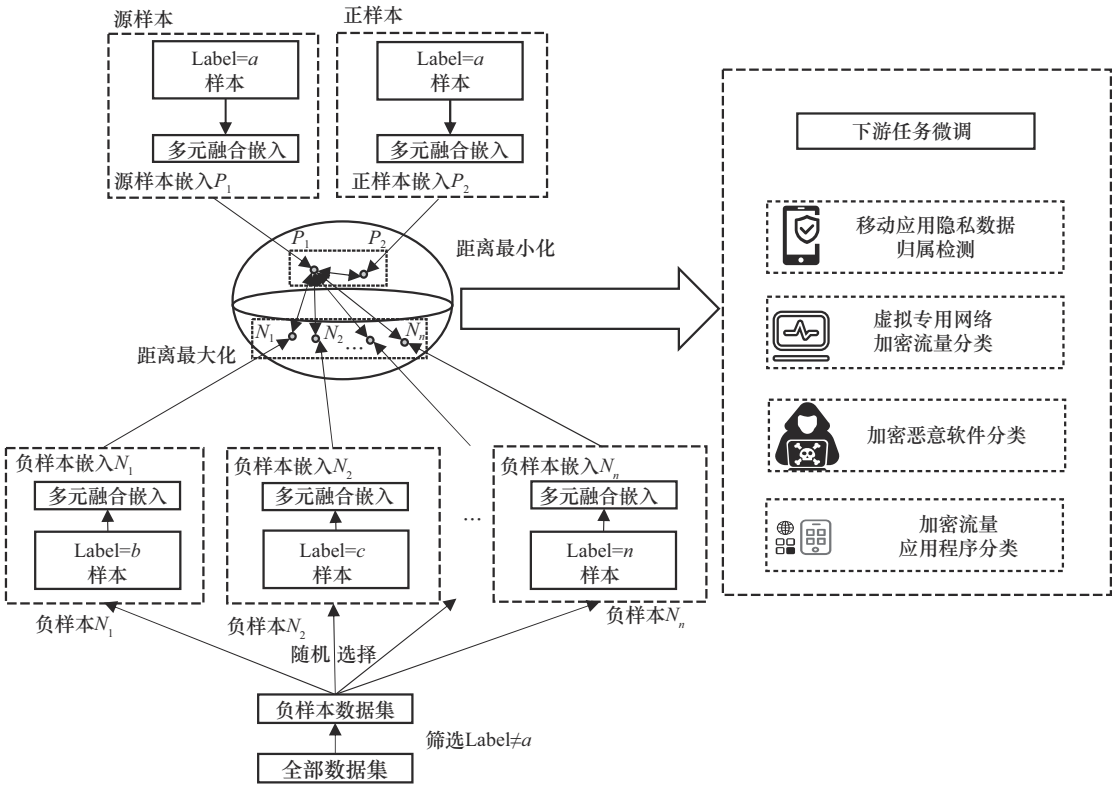

图 2　流量表征和下游任务处理过程

想对任务进行建模。主流的孪生神经网络包含2个参数相同、共享权重的子网络，对输入的一组词元的嵌入执行相同的操作，输入是源流量及其正样本或负样本组成的样本对。负样本是指在给定任务中，与目标样本（如源流量）不属于同一类别的样本。负样本的引入对于训练对比学习模型至关重要，因为它能够帮助模型区分正样本和负样本，从而提升模型在实际应用中的泛化能力。具体来说，负样本通过提供明显的区分界线，使模型能够学习到更加细致的特征表示，从而了解何种特征组合可以有效地区分源流量与不相关流量。

然而，在大规模流量检测的场景下，负样本的数量通常远大于正样本的数量。传统孪生神经网络单一负样本的输入形式，忽略了隐匿数据检测对于海量负样本差异性感知的需求。所以除原始输入外，本文提出了在引入源流量正样本的同时，引入多个源流量负样本以适应流量隐匿数据的真实分布。

图3展示了本文模型的系统结构。系统的输入为一个经过预训练模型表征的源流量、一个源流量的正样本和一组源流量的负样本，其中，正负样本数量根据经验和后验结果调优分别确认为1和9。本文使用的源流量主要是指来自国防科技大学收集到的不同用户使用的350个不同移动应用软件的网络流量。源流量的正样本是指随机挑选一个与源流量来自同一用户的隐私流量，源流量的负样本为随机挑选一组与源流量来自不同用户的隐私流量。

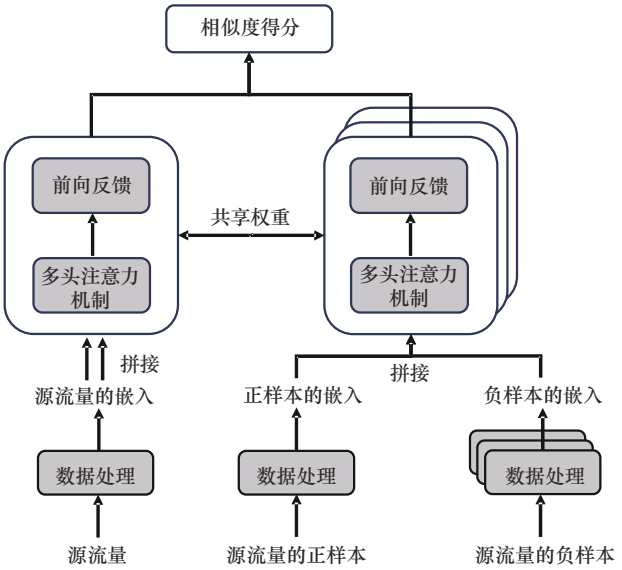

图3　基于对比学习和预训练Transformer的流量隐匿数据检测模型系统结构

在负样本选择过程中，负样本的多样性和代表性至关重要。负样本的多样性不足可能导致模型只能在某些特定类型的流量上进行训练，使模型难以应对潜在和未知的流量模式，在实际应用中泛化能力较差。此外，负样本的代表性同样至关重要，如果负样本不能有效代表真实世界中负样本的分布特征，模型可能会在训练过程中学习错误的特征组合。

在本文方法中，模型首先随机筛选一个与源流量标签不同的标签（即来自不同用户的隐私流量），然后在该标签下随机挑选一个流量作为负样本。该过程重复进行多次，以确保负样本能够充分覆盖不同用户的流量分布，从而提高模型在多样化数据上的识别能力和泛化能力。这种方法不仅能让模型学习到不同用户的流量特征，还可以为模型提供必要的区分依据，使其在处理复杂和多变的数据流时具备更强的适应能力和可靠性。

为了能够将源流量及其正负样本输入模型中，本文首先进行了Datagram2Token处理，将3组流量转换为词元嵌入的形式。随后，将其输入Transformer的编码器中，生成特征表示。再将生成的特征表示进行池化操作，得到特定需求的向量表示。最后，进行的非线性变换和线性变换。线性变换是指使用一个神经网络层对输入向量进行线性投影。非线性变换也是使用线性层将上一步的输出进一步映射到最终的目标维度。经过非线性变换和线性变换得到的输出是未归一化的概率分布，称为“logits”[38]，表示每个类别对应的分数。

为了能够将得到的logits输入扩展的孪生神经网络中进行相似值的计算，本文引入了软标签的概念。首先，将源logits复制一份，并将这2个相同的内容按维度0（行）进行拼接，形成一个更大的张量。同时将一个正样本表征的logits的第0列和n个负样本表征的logits的第0列进行拼接，形成一个新的张量。最后，将拼接后的2个张量输入扩展的孪生神经网络中，依次计算它们的余弦相似度（cosine similarity）。当2个输入经过孪生神经网络后，得到的是它们各自的特征向量，记作$f(x_1)$和$f(x_2)$，其中x_1和x_2表示2个输入，$f(\cdot)$表示共享网络生成的特征表示。此时，2个特征向量$f(x_1)$和$f(x_2)$代表了输入x_1和x_2在特征空间中的位置。然后，本文使用余弦相似度衡量2个特征向量之间

的相似性，其值域为[−1,1]，值越接近 1，表示 2 个特征向量越相似，计算式为

$$\text{cosine}_{\text{similarity}}(f(x_1),f(x_2)) = \frac{f(x_1)f(x_2)}{\|f(x_1)\|\|f(x_2)\|} \quad (2)$$

为了训练扩展的孪生神经网络，本文提出使用对比损失函数让网络逐渐学习到如何使相似样本表征对的距离更近、不相似样本表征对的距离更远，经过训练，扩展的孪生神经网络能够在特征空间中对任意 2 个输入的相似度进行更有效的判断。在本文中，为了达到这一目标，首先对每一组输入的源流量及其正负样本对的表征使用信息对比估计损失（Info_NCE, information noise contrastive estimation）函数来计算损失值，然后通过对多个源数据的损失值进行平均得到最终的损失值来训练整个网络。信息对比估计损失函数通过最大化正样本对之间相似度，最小化负样本对之间相似度的方式学习流量数据的有效表征。与交叉熵等传统损失函数相比，本文引入的损失函数能够有效地利用无监督学习的方式，从大量未标注的流量数据中学习到有用的特征表示。此外，本文引入的损失函数不仅关注正样本对的相似度，还通过负样本对的对比来优化模型的特征学习。由于其本质上是对负样本分布进行鲁棒优化而非固定分布下的优化损失函数，这种鲁棒优化方式使模型在面对负样本噪声或真实负样本难以获取的情况下，能够更稳定地学习，降低了采样误差的影响。在流量隐匿数据检测中，这种对负样本的鲁棒优化能力尤为重要，因为流量数据中可能包含大量的噪声和异常样本。信息对比估计损失的计算过程如式(3)所示。

$$L_{\text{Info\_NCE}} = -\log \frac{\exp\left(\frac{\text{sim}(\boldsymbol{q},\boldsymbol{k}^+)}{\tau}\right)}{\exp\left(\frac{\text{sim}(\boldsymbol{q},\boldsymbol{k}^+)}{\tau}\right) + \sum_{i=1}^{N}\exp\left(\frac{\text{sim}(\boldsymbol{q},\boldsymbol{k}^-)}{\tau}\right)} \quad (3)$$

其中，$\boldsymbol{q}$ 为查询向量，即输入样本表征的 logits 表示；$\boldsymbol{k}^+$为正样本表征向量，即与查询向量相关联的正样本表征的 logits 表示；$\boldsymbol{k}^-$为负样本表征向量，即与查询向量无关的负样本表征的 logits 表示；τ 为温度超参数，用于调节相似度的缩放。根据经验，本文设置温度超参数为 0.1。

3　实验

本节通过多个流量领域的主流任务，验证了模型在移动应用隐私数据归属检测、VPN 加密流量分类、加密恶意软件分类以及加密流量应用程序分类任务下的性能，然后将本文模型与 9 种不同的方法进行比较，最后对本文模型的关键组成部分进行了消融实验分析。

3.1　实验设置

3.1.1　实验数据

本文从移动应用隐私数据归属检测、VPN 加密流量分类、加密恶意软件分类以及加密流量应用程序分类任务共 4 个不同的任务上评估模型性能，每个任务的数据包数量和数据种类如表 1 所示。接下来，分别对各个任务和使用的数据集进行介绍。

表 1　数据集统计信息

数据集	数据包数量/个	数据种类/种
Mobile Traffic	245 000	49
ISCX-VPN-Service	60 000	12
ISCX-VPN-App	77 163	17
USTC-TFC	97 115	20
CSTNET-TLS 1.3	581 709	120

1) 移动应用隐私数据归属检测任务。使用国防科技大学陈曙晖教授团队收集公开的 Mobile Traffic 数据集[39]，该数据集由超过 350 个应用程序的流量组成，其中每个应用程序的流量大小不小于 100 MB，这些数据是由 224 名用户使用 10 种不同手机型号收集得到。为保障用户的隐私，该数据集进行了 UDP 报文的流量过滤，包括只保留报文前 1 500 个字节的有效载荷和个人信息匿名化（包括 IP 信息匿名化和个人信息抹除）。本文随机选择了其中 49 种类别的应用类型，每个类别最大选择 5 000 个样本。

2) VPN 加密流量分类任务。使用的数据集是 ISCX-VPN[40]，由 VPN 和非 VPN 中捕获的 6 个通信应用组成。ISCX-VPN 数据集来自加拿大 Gerard Draper-Gil 实验团队成员使用 Wireshark 和 tcpdump 捕获的团队成员产生的真实数据，包括 7 个 VPN 流量标签和 7 个非 VPN 流量标签共 14 个流量标签，如聊天通信服务、邮件服务等。数据包含源 IP 地址、源端口号、目的 IP 地址、目的端口号、传输

协议、包的大小、开始传输时间等字段。为了对本文模型在VPN加密流量的服务和应用类别上分别进行验证，本文使用包含12个标签、每个标签最大5 000个样本文件的ISCX-VPN-Service数据集和包含17个标签、每个标签最大4 500个样本文件的ISCX-VPN-App数据集。

3) 加密恶意软件分类任务。本文使用的数据集为USTC-TFC[41]。数据集包括研究人员从2011年至2015年在公共网站收集到的10种恶意软件流量，以及使用专业的网络流量模拟设备IXIA BPS收集的10种正常流量，其中包含简单邮件传送协议（SMTP, simple mail transfer protocol）、文件传输协议（FTP, file transfer protocol）、HTTP等协议标识和大小等字段。本文使用的数据集包含10类良性和10类恶意流量作为标签，每类标签最大包含5 000个样本。

4) 加密流量应用程序分类任务。使用的数据集为CSTNET-TLS 1.3[14]。该实验数据集是从2021年3月到7月在中国科技网（CSTNET, China science and technology network）收集的120个应用程序的流量数据，应用程序是从部署了TLS 1.3的网站排名Alexa Top-5000中获取的，通过服务器名称指示标记每个会话流。数据集的加密协议为TLS 1.3。本文使用了120类标签，每类标签最大5 000个样本文件作为数据集。

3.1.2　数据预处理

本文针对实验数据集进行了如下预处理操作。在移动应用隐私数据归属检测中，本文的研究主要关注HTTP的数据报文。本文对原始的pcap数据流进行了筛选、拆分与重组工作，将报文切割成只包含一条HTTP报文的pcap文件。由于数据集对个人信息进行了匿名化处理，本文向其添加了虚拟的个人信息，如手机号码、MAC地址、用户ID信息等。针对剩余任务的流量数据，本文删除了与传输具体内容无直接关联的地址解析协议及动态主机配置协议数据包，以及IP报头、以太网报头和TCP报头中的协议端口信息。处理后的数据流以二元语法编码的形式转换为连续双字节，经过字节对编码和多个特征融合得到了最后的多元融合嵌入，数据预处理流程如图4所示。在微调阶段，本文从所有数据集中的每个类中随机选择了最多5 000个样本文件大小的实验数据包。每个数据集按照8:1:1的比例划分为训练集、验证集和测试集。

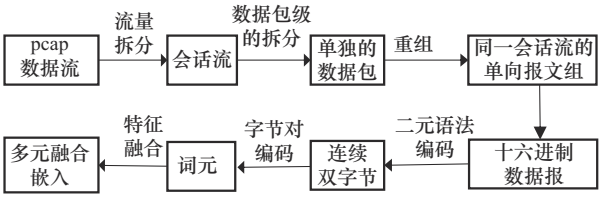

图4　数据预处理流程

3.1.3　评估指标和实现细节

本文通过4个典型指标来评估和比较本文模型的性能，包括准确率、精确率、召回率和F1分数。ACC计算方式为预测正确的所有正样本和负样本数量与总样本数量之比。PR计算方式为预测正确的正样本数量与所有预测为正样本数量之比。RC计算方式为预测正确的正样本数量与真实正样本数量之比。F1分数是PR和RC的一个加权平均值。通过计算每一类数据的ACC、PR、RC和F1分数的平均值，避免了多种类型数据之间样本数量不平衡而导致的实验结果偏差。

在预训练阶段，本文设置批大小为32，学习率为2×10^{-5}，预训练比率为0.1。本文设置总步长根据不同任务进行调整，并使用AdamW优化器对10个epoch进行微调，实验基于Pytorch 1.8.0和预训练框架UER（universal encoder representation）实现，使用2个型号为NVIDIA Corporation TU104GL的图形计算单元（GPU, graph process unit）。

3.2　移动应用隐私数据归属检测对比实验

在移动应用隐私数据归属检测任务中，将本文方法与ET-BERT方法进行比较，选取的实验数据为国防科技大学Mobile Traffic数据集，随机选取49位用户的流量信息作为本文数据集，故实验标签数量为49，为对比本文方法在不同数据集上的表现，本文将样本数量分别设置为1 000和3 000。

移动应用隐私数据归属检测对比实验结果如图5所示。由图5可知，本文方法显著优于ET-BERT方法。在样本数量为1 000时，本文方法相较于ET-BERT方法在准确率方面显著提高9.3%，在精确率显著提高了9.28%。同时在召回率和F1分数上分别有3.24%和6.58%的提升。在样本数量为3 000时，本文方法相较于ET-BERT方法在准确率方面显著提高5.27%，在精确率方面显著提高了

5.51%。同时在召回率和F1分数上分别有0.48%和3.12%的提升。

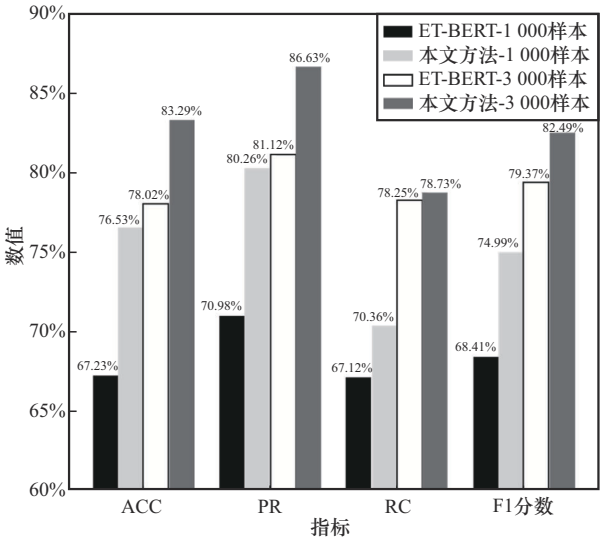

图5　移动应用隐私数据归属检测对比实验结果

实验结果表明，本文方法在移动应用隐私数据归属检测任务中更具优势，通过在训练过程中使用更多的负样本，提高了模型训练的有效性。

3.3　模型泛化能力对比实验

为验证本文方法在不同任务上的通用性，本文调研了当前加密流量分类领域的前沿方法，将模型与9种主流方法进行比较：1) 指纹构建法FlowPrint[42]；2) 基于统计特征的检测方法AppScanner[28]和BIND[43]；3) 基于深度学习的检测方法DF（deep fingerprinting）[30]、FS-Net[13]、GraphDApp[44]和Deeppacket[19]；4) 基于预训练模型的检测方法PERT[12]和ET-BERT[14]。实验结果如表2所示。在VPN加密流量分类任务中，从图6的结果可以看出，本文方法在ISCX-VPN-Service数据集上比现有方法在ACC、PR和F1分数上均有提升。从图7可以看出，本文方法在ISCX-VPN-App数据集的4项指标均优于其他方法，在ACC上更是达到了99.68%。这2个数据集都提出了数据不平衡的挑战，本文方法和ET-BERT方法通过学习分组数据报之间的相关性来缓解数据不平衡的影响。具体来说，本文通过向模型提供更多的负样本对使模型可以学习到更多不同标签的区别，进而获得更好的差异性检测能力。实验结果表明，本文方法不仅能够在多个主流加密流量的下游任务上取得较优效果，同时，即使在数据不平衡的情况下也具有较强的识别加密流量隐匿数据的能力。

USTC-TFC数据集实验结果如表2所示。参与对比的模型在该实验数据集上都有较为优异的表现。通过对数据集的进一步分析，该数据集中的恶意流量在应用层中包含部分明文信息，这使各类模型可以通过未加密的数据信息进行分类识别，从而获得了较好的性能。尽管如此，从图8可以看出，本文方法的表现依然明显优于其他对比方法。本文

表2　模型在VPN加密流量分类、加密恶意软件分类与加密流量应用程序分类的泛化能力对比实验结果

方法	ISCX-VPN-Service				ISCX-VPN-App				USTC-TFC				CSTNET-TLS 1.3			
	ACC	PR	RC	F1分数	ACC	PR	RC	F1分数	ACC	PR	RC	F1分数	ACC	PR	RC	F1分数
AppScanner	71.82%	73.39%	72.25%	71.97%	62.66%	48.64%	51.98%	49.35%	89.54%	89.84%	89.68%	88.92%	66.62%	62.46%	63.27%	62.01%
BIND	75.34%	75.83%	74.88%	74.20%	67.67%	51.52%	51.53%	49.65%	84.57%	86.81%	83.82%	83.96%	79.64%	76.05%	76.50%	75.60%
FlowPrint	79.62%	80.42%	78.12%	78.20%	87.67%	66.97%	66.51%	65.31%	81.46%	64.34%	70.02%	65.73%	12.61%	13.54%	12.72%	11.16%
DF	71.54%	71.92%	71.04%	71.02%	61.16%	57.06%	47.52%	47.99%	77.87%	78.83%	78.19%	75.93%	79.36%	77.21%	75.73%	76.02%
FS-Net	72.05%	75.02%	72.38%	71.31%	66.47%	48.19%	48.48%	47.37%	88.46%	88.46%	89.20%	88.40%	86.39%	84.04%	83.49%	83.22%
GraphDApp	59.77%	60.45%	62.20%	60.36%	63.28%	59.00%	54.72%	55.58%	87.89%	82.26%	82.60%	82.34%	70.34%	64.64%	65.10%	64.40%
Deeppacket	93.29%	93.77%	93.06%	93.21%	97.58%	97.85%	97.45%	97.65%	96.40%	96.50%	96.31%	96.41%	80.19%	43.15%	26.89%	40.22%
PERT	93.52%	94.00%	93.49%	93.68%	82.29%	70.92%	71.73%	69.92%	99.09%	99.11%	99.10%	99.11%	89.15%	88.46%	87.19%	87.41%
ET-BERT	98.90%	98.91%	**98.90%**	98.90%	99.62%	99.36%	99.38%	99.37%	99.15%	99.15%	99.16%	99.16%	97.37%	97.42%	**97.42%**	97.41%
本文方法	**99.02%**	**99.76%**	98.28%	**99.02%**	**99.68%**	**99.96%**	**99.40%**	**99.68%**	**99.39%**	**99.39%**	**99.39%**	**99.39%**	**97.68%**	**97.97%**	97.23%	**97.60%**

方法以高达99.39%的ACC和99.39%的F1分数优于对比方法中表现最好的ET-BERT方法。实验结果表明，本文方法在加密恶意软件分类任务上各项指标的表现较以往方法均有进一步提升。

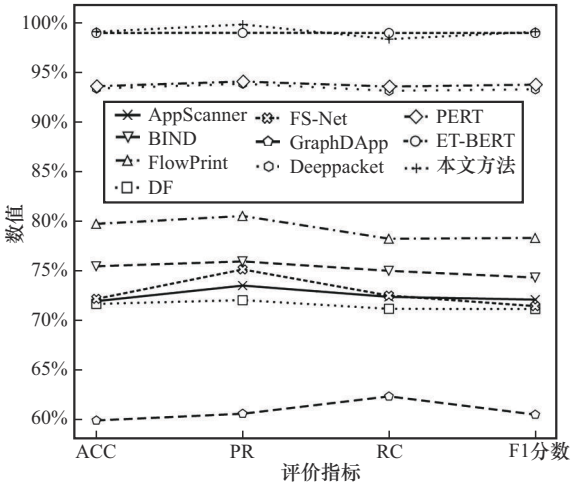

图6　VPN加密流量分类ISCX-VPN-Service数据集对比实验结果

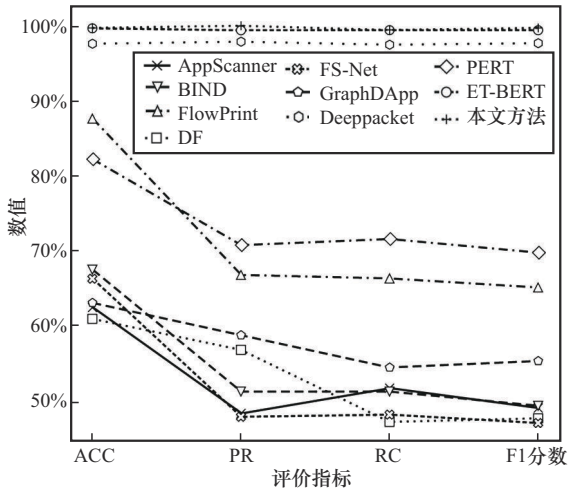

图7　VPN加密流量分类ISCX-VPN-App数据集对比实验结果

在基于CSTNET-TLS 1.3数据集的加密流量应用程序分类任务中，不同方法在这一任务上表现差异较为明显，如图9所示。其中，本文方法在ACC上取得了最佳效果，达到了97.68%。同时在PR和F1分数上也取得了最优异的效果，分别为97.97%和97.60%。但是本文方法在RC这一指标上略低于ET-BERT方法。本文方法通过输入一个正样本对和多个负样本对的方式促进模型对不同标签特征的学习，忽略了正样本的特征学习，致使模型出现了误判，导致在RC这一指标上效果略低于ET-BERT方法。

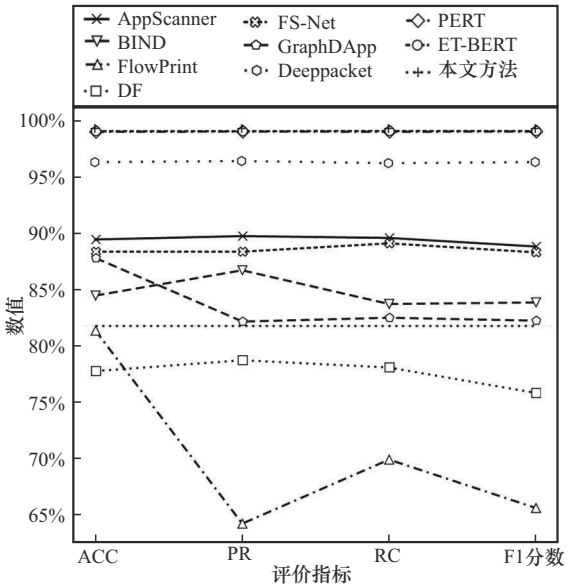

图8　加密恶意软件分类USTC-TFC数据集对比实验结果

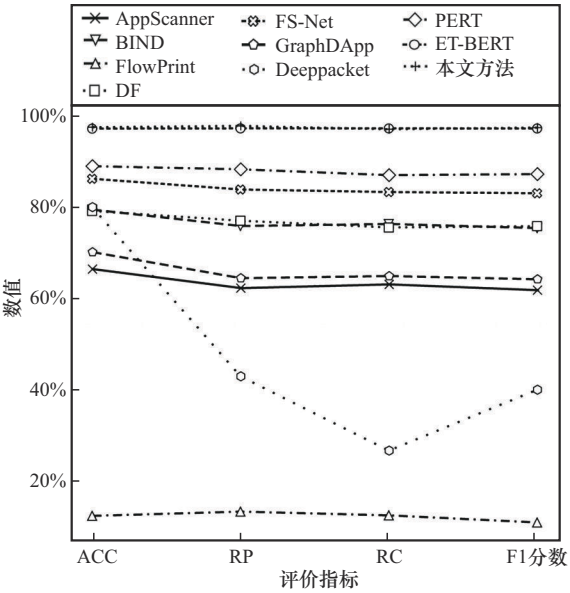

图9　加密流量应用程序分类CSTNET-TLS 1.3数据集的对比实验结果

3.4　消融实验

本文的消融实验旨在探讨预训练的嵌入和基于对比学习微调范式这2项优化对实验结果的量化增益。为此，实验设置如下方案：1) 方案1为输入中不包含正负样本，仅使用预训练Transformer模型得到流量嵌入；2) 方案2使用源流量及其一个正样本和负样本作为输入的孪生神经网络得到流量嵌入；3) 方案3为本文方法，同时引入预训练嵌入和对比学习方法，每次使用源流量及其一个正样本和多个负样本作为输入，评估二者联合对实验结果的增益效果。本文选取Mobile Traffic作为数据集，

标签数量为 49，样本数量为 3 000，实验结果如表 3 所示。

表 3　本文方法关键组成部分的消融实验结果

方案	ACC	PR	RC	F1 分数
方案 1	78.02%	81.12%	78.25%	79.36%
方案 2	79.13%	80.74%	76.51%	78.57%
方案 3（本文方法）	83.36%	86.63%	78.73%	82.49%

在方案 1 中，模型的 ACC 值达到了 78.02%，PR 值为 81.12%，RC 值为 78.25%，F1 值为 79.36%。方案 2 通过结合对比学习的方式相较于方案 1 在 ACC 上提升了 1.11%，为 79.13%。但方案 2 在 PR 上较方案 1 降低了 0.38%，为 80.74%；在 RC 上较方案 1 降低了 1.74%，为 76.51%；在 F1 分数上较方案 1 降低了 0.79%，为 78.57%。这表明对比学习方法虽然略微提高了分类器区分正确类别的能力，但只引入一个负样本的对比学习可能使模型过于专注于给定负样本的特征，从而忽略了其他重要信息，特别是在多类别分类任务中，引入了更多的误报和漏报，故而造成某些类别的召回率下降，导致模型整体性能并未得到全面提升。

本文方法在 ACC、PR、RC 和 F1 分数 4 项指标上都取得了最佳的效果，分别为 83.36%、86.63%、78.73% 和 82.49%。相比方案 1，本文方法的准确率提升了 5.34%；相比方案 2，本文方法的准确率提升了 4.32%。这一显著提升说明扩展孪生神经网络结构在正确分类样本方面具有更强的能力。在精确率方面，本文方法相较于方案 1 提升了 5.51%，相较于方案 2 提升了 5.89%。这表明模型在预测正类时的准确性更高，减少了误报情况。尽管本文方法的召回率略有波动，但仍保持在 78.73%，较方案 1 提升了 0.48%，较方案 2 提升了 2.22%。这说明模型在识别所有正类样本方面的表现有所改善。F1 分数是精确率和召回率的调和平均数，本文方法达到了 82.49%，较方案 1 提升了 3.13%，较方案 2 提升了 3.92%，体现了模型在精确率和召回率之间的平衡性得到了优化。

4　讨论

4.1　本文方法的潜在局限

在进行加密流量的同源性分析中，本文选择了 Mobile Traffic 数据集，该数据集由国防科技大学实验团队开发的应用程序 Netlog 收集。在对该数据集进行筛选切割时，无法使用较为常用的 Splitcap 应用程序。本文选择使用 Tshark 应用程序对该数据集进行 HTTP 报文筛选，然后使用 Editcap 应用程序进行报文切割。由于 Editcap 应用程序只能完成 packet 层级的切割，无法进行 flow 层级的切割，一定程度上限制了本文方法的应用。另外，不同用户产生的 HTTP 报文数量较为不平衡。为实现大规模数据上的实验比较，本文对数据种类进行了限制，移除了样本数量少于 5 000 的流量数据，最终得到了 49 个类别的流量数据，参考 ET-BERT 等工作的流量类别数量，本文认为 49 个类别作为移动应用隐私数据归属检测的种类是充分的。

4.2　个人信息生成规则

Mobile Traffic 数据集对所有可能的个人信息（如用户 ID、设备 ID、MAC 地址、位置信息、邮件以及手机号码）进行了匿名处理，将所有的用户 IP 地址替换为 10.1.10.1 的虚拟 IP 地址。为此，本文制定了如下的规则以实现个人信息的模拟。

首先是用户的 IP，原数据集将每位用户的 IP 地址统一隐藏为 10.1.10.1。本文为不同的标签生成符合规则的随机化 IP 地址，随后将每一个文件里的每一条报文中的隐藏 IP 替换为随机 IP。然后是用户的手机号码，本文为同一用户生成固定的 11 位手机号码，并按原数据集的匿名规则，在长度介于 21~60 的所有连续空白字符串中反向添加字符串“phone_num=”标识和手机号。接下来是用户 ID，本文通过与大规模应用程序交互后发现，报文里的用户 ID 长度并不固定，最小为 9 位，最大为 37 位。本文选取出现次数最多的 32 位作为标准，为同一用户生成固定的 32 位用户 ID，并在字符串前方添加“user_id=”标识。随后匹配报文中长度介于 61~105 的连续空白内容，填入手机号和用户 ID。最后是用户的 MAC 地址，本文随机生成 32 位的 MAC 地址，并在字符串前方添加“mac_address=”标识。本文将报文中长度大于 109 的连续空白内容填入手机号、用户 ID 和 MAC 地址。综上所述，本文通过虚拟个人数据的生成，在不侵犯个人隐私的前提下，满足了移动应用隐私数据归属检测任务的数据种类和样本数量的要求。

5　结束语

本文针对大规模流量报文数据，提出了一种差异性敏感的隐匿数据检测方法，其基于预训练

Transformer模型和对比学习的框架能够高效实现移动应用隐私数据归属检测、VPN加密流量分类、加密恶意软件分类和加密流量应用程序分类任务。本文在4个公开数据集上与9个主流方法进行了4种不同任务的对比实验，结果表明在移动应用隐私数据归属检测任务上，本文方法相较于主流方法在准确率方面显著提高5.27%，在精确度方面显著提高5.51%。同时在召回率和F1分数上分别有0.48%和3.12%的提升。此外，在加密恶意软件分类、加密流量应用程序分类以及VPN加密流量分类任务上，本文方法分别达到了99.39%、97.68%以及99.68%的准确率，优于现有的对比方法。在未来，计划研究针对流量领域预训练模型的对抗攻击手段，预防潜在恶意攻击者对检测系统的攻击和绕过。

参考文献：

[1] REZAEI S, LIU X. Deep learning for encrypted traffic classification: an overview[J]. IEEE Communications Magazine, 2019, 57(5): 76-81.

[2] BARTOS K, SOFKA M, FRANC V. Optimized invariant representation of network traffic for detecting unseen malware variants[C]//Proceedings of the 25th USENIX Conference on Security Symposium. New York: ACM Press, 2016: 807-822.

[3] WANG S S, YAN Q B, CHEN Z X, et al. Detecting Android malware leveraging text semantics of network flows[J]. IEEE Transactions on Information Forensics and Security, 2017, 13(5): 1096-1109.

[4] ROESCH M. Snort-lightweight intrusion detection for networks[C]// Proceedings of the 13th USENIX Conference on System Administration. New York: ACM Press, 1999: 229-238.

[5] ANDERSON B, MCGREW D. Identifying encrypted malware traffic with contextual flow data[C]//Proceedings of the 2016 ACM Workshop on Artificial Intelligence and Security. New York: ACM Press, 2016: 35-46.

[6] TEGELER F, FU X M, VIGNA G, et al. BotFinder: finding bots in network traffic without deep packet inspection[C]//Proceedings of the 8th International Conference on Emerging Networking Experiments and Technologies. New York: ACM Press, 2012: 349-360.

[7] XIE R J, WANG Y X, CAO J H, et al. Rosetta: enabling robust TLS encrypted traffic classification in diverse network environments with TCP-aware traffic augmentation[C]//Proceedings of the ACM Turing Award Celebration Conference-China 2023. New York: ACM Press, 2023: 131-132.

[8] 潘吴斌, 程光, 郭晓军, 等. 网络加密流量识别研究综述及展望[J]. 通信学报, 2016, 37(9): 154-167.
PAN W B, CHENG G, GUO X J, et al. Review and perspective on encrypted traffic identification research[J]. Journal on Communications, 2016, 37(9): 154-167.

[9] ZHENG W B, GOU C, YAN L, et al. Learning to classify: a flow-based relation network for encrypted traffic classification[C]//Proceedings of the Web Conference 2020. New York: ACM Press, 2020: 13-22.

[10] 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. Mexico: Association for Computational Linguistics, 2019: 4171-4186.

[11] VASWANI A, SHAZEER N, PARMAR N, et al. Attention is all you need[C]//Proceedings of the 31st International Conference on Neural Information Processing Systems. 2017: 6000-6010.

[12] HE H Y, YANG Z G, CHEN X N. PERT: payload encoding representation from transformer for encrypted traffic classification[C]//Proceedings of the 2020 ITU Kaleidoscope: Industry-Driven Digital Transformation (ITU K). Piscataway: IEEE Press, 2020: 1-8.

[13] LIU C, HE L T, XIONG G, et al. FS-Net: a flow sequence network for encrypted traffic classification[C]//Proceedings of the IEEE INFOCOM 2019-IEEE Conference on Computer Communications. Piscataway: IEEE Press, 2019: 1171-1179.

[14] 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 Press, 2022: 633-642.

[15] CHEN Z, CHENG G, XU Z. A survey on Internet encrypted traffic detection classification and identification[J]. Chinese Journal of Computers, 2023, 46(2023): 1060-1085.

[16] 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.

[17] WANG P, LI S H, YE F, et al. PacketCGAN: exploratory study of class imbalance for encrypted traffic classification using CGAN[C]//Proceedings of the ICC 2020-2020 IEEE International Conference on Communications (ICC). IEEE, 2020: 1-7.

[18] ACETO G, CIUONZO D, MONTIERI A, et al. Mobile encrypted traffic classification using deep learning: experimental evaluation, lessons learned, and challenges[J]. IEEE Transactions on Network and Service Management, 2019, 16(2): 445-458.

[19] LOTFOLLAHI M, SIAVOSHANI M J, ZADE R S H, et al. Deep packet: a novel approach for encrypted traffic classification using deep learning[J]. Soft Computing, 2020, 24(3): 1999-2012.

[20] LI R, XIAO X, NI S G, et al. Byte segment neural network for network traffic classification[C]//Proceedings of the 2018 IEEE/ACM 26th International Symposium on Quality of Service (IWQoS). Piscataway: IEEE Press, 2018: 1-10.

[21] AZAB A, KHASAWNEH M, ALRABAEE S, et al. Network traffic classification: techniques, datasets, and challenges[J]. Digital Communications and Networks, 2024, 10(3): 676-692.

[22] SHAFIQ M, TIAN Z H, BASHIR A K, et al. Data mining and machine learning methods for sustainable smart cities traffic classification: a survey[J]. Sustainable Cities and Society, 2020, 60: 102177.

[23] GANAIE M A, HU M H, MALIK A K, et al. Ensemble deep learning: a review[J]. Engineering Applications of Artificial Intelligence, 2022, 115: 105151.

[24] ACETO G, CIUONZO D, MONTIERI A, et al. DISTILLER: encrypted traffic classification via multimodal multitask deep learning[J].

Journal of Network and Computer Applications, 2021, 183: 102985.

[25] YUN X C, WANG Y P, ZHANG Y Z, et al. Encrypted TLS traffic classification on cloud platforms[J]. IEEE/ACM Transactions on Networking, 2023, 31(1): 164-177.

[26] DONG S, WANG P, ABBAS K. A survey on deep learning and its applications[J]. Computer Science Review, 2021, 40: 100379.

[27] 郭帅, 苏旸. 基于数据流的加密流量分类方法[J]. 计算机应用, 2021, 41(5): 1386-1391.
GUO S, SU Y. Encrypted traffic classification method based on data stream[J]. Journal of Computer Applications, 2021, 41(5): 1386-1391.

[28] 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.

[29] SENGUPTA S, GANGULY N, DE P, et al. Exploiting diversity in Android TLS implementations for mobile app traffic classification[C]// Proceedings of the World Wide Web Conference. New York: ACM Press, 2019: 1657-1668.

[30] 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.

[31] ZENG Y, GU H X, WEI W T, et al. Deep-full-range: a deep learning based network encrypted traffic classification and intrusion detection framework[J]. IEEE Access, 2019, 7: 45182-45190.

[32] JIANG S, FU C, QIAN Y K, et al. IFAttn: binary code similarity analysis based on interpretable features with attention[J]. Computers & Security, 2022, 120: 102804.

[33] LIU J M, FU Y J, MING J C, et al. Effective and real-time in-app activity analysis in encrypted Internet traffic streams[C]//Proceedings of the 23rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. New York: ACM Press, 2017: 335-344.

[34] WILLINGER W, TAQQU M S, LELAND W E, et al. Self-similarity in high-speed packet traffic: analysis and modeling of Ethernet traffic measurements[J]. Statistical Science, 1995, 10(1): 67-85.

[35] CHUNG Y A, BELINKOV Y, GLASS J. Similarity analysis of self-supervised speech representations[C]//Proceedings of the ICASSP 2021-2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP). IEEE, 2021: 3040-3044.

[36] 何高峰, 魏千峰, 肖咸财, 等. 支持数据隐私保护的恶意加密流量检测确认方法[J]. 通信学报, 2022, 43(2): 156-170.
HE G F, WEI Q F, XIAO X C, et al. Confirmation method for the detection of malicious encrypted traffic with data privacy protection[J]. Journal on Communications, 2022, 43(2): 156-170.

[37] CHEN X L, HE K M. Exploring simple Siamese representation learning[C]//Proceedings of the 2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). IEEE, 2021: 15750-15758.

[38] ZHANG J, LI Z Q, LI B, et al. Federated learning with label distribution skew via logits calibration[C]//Proceedings of the International Conference on Machine Learning (LCML). Maryland: International Machine Learning Society, 2022: 26311-26329.

[39] ZHAO S, CHEN S H, WANG F, et al. A large-scale mobile traffic dataset for mobile application identification[J]. The Computer Journal, 2024, 67(4): 1501-1513.

[40] DRAPER-GIL G, LASHKARI A H, MAMUN M S I, et al. Characterization of encrypted and VPN traffic using time-related features[C]// Proceedings of the 2nd International Conference on Information Systems Security and Privacy. SCITEPRESS-Science and Technology Publications, 2016: 407-414.

[41] WANG W, ZHU M, ZENG X W, et al. Malware traffic classification using convolutional neural network for representation learning[C]//Proceedings of the 2017 International Conference on Information Networking (ICOIN). Piscataway: IEEE Press, 2017: 712-717.

[42] EDE T V, 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. Internet Society, 2020: 1-18.

[43] AL-NAAMI K, CHANDRA S, MUSTAFA A, et al. Adaptive encrypted traffic fingerprinting with bi-directional dependence[C]//Proceedings of the 32nd Annual Conference on Computer Security Applications. New York: ACM Press, 2016: 177-188.

[44] SHEN M, ZHANG J P, ZHU L H, et al. Accurate decentralized application identification via encrypted traffic analysis using graph neural networks[J]. IEEE Transactions on Information Forensics and Security, 2021, 16: 2367-2380.

[作者简介]

何帅（1994– ），男，新疆乌鲁木齐人，华中科技大学博士生，主要研究方向为流量安全、软件安全、人工智能安全等。

张京超（2001– ），男，河北邯郸人，华中科技大学硕士生，主要研究方向为流量安全、二进制代码分析等。

徐笛（2000– ），女，河北衡水人，华中科技大学硕士生，主要研究方向为流量安全、二进制代码逆向分析等。

江帅（1996– ），男，重庆人，华中科技大学博士生，主要研究方向为二进制代码分析、漏洞分析等。

郭晓威（1996– ），男，广西贺州人，华中科技大学博士生，主要研究方向为数据安全、源代码作者溯源、恶意代码检测技术等。

付才（1976– ），男，湖北通城人，博士，华中科技大学教授、博士生导师，主要研究方向为网络行为分析、移动网络安全、恶意代码等。