Computer Science and Application 计算机科学与应用, 2022, 12(5), 1425-1435
Published Online May 2022 in Hans. http://www.hanspub.org/journal/csa
https://doi.org/10.12677/csa.2022.125142

Hans汉斯

基于图嵌入和多模态深度学习的加密流量分类

杨瑞鹏[1,2]，于爱民[1]，蔡利君[1]，孟　丹[1]

[1]中国科学院信息工程研究所，北京
[2]中国科学院大学网络空间安全学院，北京

收稿日期：2022年4月26日；录用日期：2022年5月24日；发布日期：2022年5月31日

摘　要

伴随着互联网的发展，新的应用层出不穷，并且加密流量所占流量比例不断提高。与此同时，多数网络恶意攻击也以加密的形式在网络中传播。因此，对网络流量进行精细化分类有利于提高网络管理水平和减少网络安全风险。传统的基于端口和基于负载的分类方法对海量应用和加密流量已经不再适用，使用机器学习的加密流量分类方法的性能受限于流量的统计特征。在本文中，我们提出了基于图嵌入和多模态深度学习的加密流量分类方法。该方法使用多模态深度学习模型联合了两种类型的特征——流序列特征和图嵌入特征。实验结果表明，我们所提出的方法明显优于已有最先进的方法，并且具有很好的抗干扰能力。

关键词

图嵌入，加密流量分类，深度学习

Encrypted Traffic Classification Based on Graph Embedding and Multimodal Deep Learning

RuipengYang[1,2], Aimin Yu[1], Lijun Cai[1], Dan Meng[1]

[1]Institute of Information Engineering, Chinese Academy of Sciences, Beijing
[2]School of Cyber Security, University of Chinese Academy of Sciences, Beijing

Received: Apr. 26th, 2022; accepted: May 24th, 2022; published: May 31st, 2022

Abstract

With the rapid development of the Internet, more and more new applications are emerging, and

文章引用: 杨瑞鹏, 于爱民, 蔡利君, 孟丹. 基于图嵌入和多模态深度学习的加密流量分类[J]. 计算机科学与应用, 2022, 12(5): 1425-1435. DOI: 10.12677/csa.2022.125142

the proportion of encrypted traffic continues to increase. At the same time, most network malicious attacks are also spread in the network in encrypted form. Therefore, the refined classification of network traffic is compulsory to improve the efficiency of the network and reduce network security risks. Traditional port-based and load-based classification methods are no longer applicable to massive applications and encrypted traffic, and the performance of encrypted traffic classification methods using machine learning is limited by the statistical characteristics of traffic. In this paper, we propose an encrypted traffic classification method based on graph embedding and multimodal deep learning. The method combines two types of features—flow sequence features and graph embedding features using a multimodal deep learning model. Experimental results show that our proposed method significantly outperforms the existing state-of-the-art methods and has a good anti-interference ability.

Keywords

Graph Embedding, Encrypted Traffic Classification, Deep Learning

1. 引言

近年来，随着互联网技术的飞速发展和用户规模不断增加，各种网络应用应运而生。一方面，人们利用网络进行购物、社交、娱乐、学习、办公、支付、炒股……；另一方面，很多企业和服务机构也利用网络开展业务，例如发布信息、招聘员工、网上办理业务等活动。这就导致网络中的流量和应用呈爆发式增长，使网络更加难以管理。

互联网为我们提供便捷服务的同时，也带来了网络安全问题，例如网络上的信息被非法监听、破坏、修改等。为了信息的安全、可靠传输，人们使用加密技术来对网络中的流量进行加密，这就使得加密流量在网络流量中的占比越来越高。并且，很多恶意软件的流量也以加密的形式在网络中传播，进行破坏。

通过对网络流量进行分类，不仅可以帮助网络监管者进行网络流量监控、网络带宽管理、提高服务质量，还可以及时发现网络攻击、对异常流量进行检测。因此，对网络流量进行分类有助于提高网络管理水平和网络安全水平。

早期人们通过分析端口和负载来分类流量[1]-[8]。由于许多应用程序使用动态端口，或者通过端口混淆技术伪装成某些熟知的端口号，因此基于端口的方法准确率变得越来越低。基于负载的方法对网络数据包的内容进行分析，提取特征串，使用特征匹配算法来分类流量，但是，它只对明文有效，无法解析加密流量。因此，基于端口和基于负载的分类方法不再适用于当今的加密流量。

基于机器学习的加密流量分类方法[9] [10] [11] [12]通过分析网络流的统计信息来对流量进行分类，例如，网络流持续时间的分布特性、包间隔时间、包长特征等。自 2004 年至今，有多种机器学习算法被用来分析网络流。2016 年，M. Conti [13]首先使用 DTW (动态时间规整算法)来计算流时间间隔序列的相似度，然后通过对包长度进行聚类来分析 SSL 加密流量，用来推断手机 Android 用户的行为。V. F. Taylor [14]提出了一个 AppScanner 的框架来识别加密网络流量中的 Android 应用，文章中作者定义了流量爆发的概念把数据包分割成不同流，然后把数据流分成基于统计和包长度两组特征，再使用机器学习对 Google Play 中的 110 个最受欢迎的应用进行识别，达到了很高的识别率。B. Anderson [15] [16]在 2016 年利用背

景流量数据来识别 TLS 加密恶意流量，他首先分析了百万级的正常流量和恶意流量中 TLS 流、DNS 流和 HTTP 流的不同之处，然后选取具有明显区分度的特征集作为分类器(有监督机器学习)的输入来训练检测模型，从而识别加密的恶意流量。在 2017 年，B. Anderson [17]又比较了不同分类器在识别恶意加密流量的性能，得出随机森林算法优于其他机器学习算法(逻辑回归、决策树、SVM 等)。

虽然基于机器学习的方法准确率较高，应用范围很广，但是也有一些不足之处。比如特征重叠问题，不同的网络流可能具有相似的特征，这就会给网络流分类带来干扰。还有随着流量识别技术的进步，一些反流量识别技术(如流量伪装、流量特征模糊)也在发展。这给加密流量分类带来了困难。

在本文中，我们为加密流量分类问题提出了一种基于图嵌入和多模态深度学习的模型。该模型使用多模态深度学习模型联合了两种类型的特征——流序列特征和图嵌入特征。其中流序列特征指的是网络流的包长度序列，图嵌入特征是加密应用的流量表示成图结构后，通过图嵌入技术得到的低维嵌入向量。为了学习到图中节点的嵌入向量表示。首先，我们把加密应用产生的网络流表示成图结构，这种图结构能够表征加密应用的空间分布情况；然后，通过图嵌入学习到关于图中节点的低维的向量表示，这种低维向量表示不仅反映了在空间上加密应用之间的关系，而且图的结构化信息经过嵌入后，有利于计算和学习；最后，我们使用多模态深度学习模型学习加密应用的流序列特征和图嵌入特征，实验结果表明，我们所提出的方法明显优于已有最先进的方法，并且具有很好的抗干扰能力。

本文的主要贡献如下：

1) 为了提取加密应用的空间特征，提出了把加密应用的网络流表示成图的方法。

2) 使用图嵌入技术对加密应用的网络流图进行处理，得到节点的低维嵌入向量。

3) 提出了基于多模态学习的模型框架，对加密应用的嵌入向量和流序列特征进行联合训练，取得了很好的实验效果。

2. 特征提取过程

2.1. 网络流的定义和提取

网络中的数据包是构成流量的基本单位，一连串的数据包就组成了网络流，网络流量越大意味着数据包数目多或者载荷较大。原始的网络流量是具有不同 IP 地址和端口的数据包混合在一起的，我们使用五元组信息来把混乱的数据包组成流，我们的研究工作就是把这些流划分到不同的应用中去。五元组信息包括：源 IP、目的 IP、源端口、目的端口以及传输层协议(TCP 或者 UDP)，具有相同五元组信息的一连串数据包就是一条应用流。

我们这样定义流的开始和结束。当出现一个数据包或者上一条流结束的时候，就是流的开始；当流持续时间超过规定时间，或者检测到包含 RST 或者 FIN 标志的数据报文，就可以认为一条流结束。在 TCP 协议中，“RST”表示接收数据超时，“FIN”表示数据发送完毕，当这两个标志出现时，就可以认为一条流的结束。UDP 是面向无连接的协议，只能通过设置好的时间来判定流的结束。

2.2. 流序列特征

本文中，我们使用网络流中的包长度序列来作为加密应用的流序列特征。对于一个包序列 $S=\left[a_1,a_2,\cdots,a_m\right]$，$m$ 是包序列长度，a_m 表示第 m 个包。这里，我们通过一个函数 $f\left(a_i\right)$ 把序列 S 转化为序列 $S'=\left[f\left(a_1\right),f\left(a_2\right),\cdots,f\left(a_m\right)\right]$。其中，转换函数 $f\left(a_i\right)$ 定义如下：

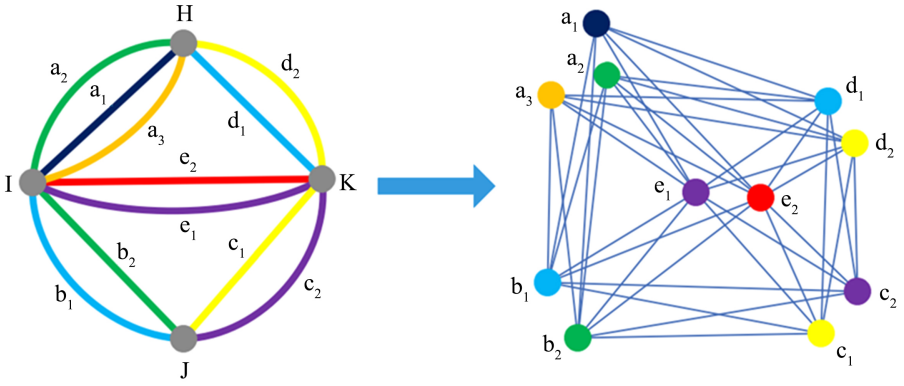
$$f\left(a_i\right)=\begin{cases}\left|a_i\right| & \text{若 } a_i \text{ 从服务器发到用户}\\ -\left|a_i\right| & \text{若 } a_i \text{ 从用户发到服务器}\end{cases} \tag{1}$$

可以看到，包长度由绝对值来表示，数据包的正负符号表示方向，正值表示从服务器到主机，负值表示从主机到服务器。

2.3. 加密应用的嵌入向量

2.3.1. 图的构造和变换

为了方便图嵌入，我们对图进行构造和变换，如图 1 所示。



(a) 构造的原始图　　(b) 变换后的图

Figure 1. Graph construction and graph transformations
图 1. 图构造和图变换

我们根据加密应用流的源 IP 地址和目的 IP 地址构造图 G (V, E)，其中顶点 V 代表 IP 地址，边 E 代表两个 IP 地址之间的数据流。这里，为了简单起见，我们不考虑流的方向，因此图是无向的。图 1(a)是我们所要构造加密应用流的图，其中顶点 H, I, J, K 代表 4 个 IP 地址，边表示 IP 对之间的流。我们知道，一条应用流包含了五元组信息，除了 IP 地址外，还包括端口号和传输层协议类型，然而，图中的顶点只包含 IP 地址信息。因此，具有相同 IP 对的流可能是多条流，我们对这些具有相同 IP 对的不同流进行了编号，两个 IP 之间编号的个数就是两个节点之间边的个数。如图所示，IP 地址 H 和 I 之间有 3 条边，这 3 条边的编号为 a_1， a_2 和 a_3，对应 3 种不同应用，这 3 种应用我们用不同颜色区分出来。IP 对和应用之间的关系是：同一个 IP 对，可能包含了不同应用；一个应用也可能具有不同的 IP 对。

图中的边表示应用流，为了方便在后续的图嵌入过程中进行处理，我们需要对构造的原始图进行变换。我们把原始图记为 $G=(V,E)$， $v\in V$， $e\in E$，把变换后的图记为 $G'=(V',E')$， $v'\in V'$， $e'\in E'$，其中 v 和 e 分别表示顶点集合 V 和边集合 E 中的顶点和边， v' 和 e' 类似。

我们通过以下 3 个步骤，来对图进行变换。

1) 记录下图 G 中每条边的相邻边集合。设边 e_x 的相邻边构成了集合记为 E_x，计算所有边 $e_1,e_2\cdots,e_n$ 的相邻边集合 $E_1,E_2\cdots,E_n$。边 e_x 和相邻边集合 E_x 的数学化表示为：

$$\begin{cases} e_x=(v_i,v_j)\in E \\ E_x=\{(u_i,v_i)\in E\,|\,u_i\in V,u_i\neq v_j\}+\{(u_j,v_j)\in E\,|\,u_j\in V,u_j\neq v_i\} \end{cases} \tag{2}$$

2) 把图 G 中的边 e_x 收缩成顶点，成为图 G' 中的 v'_x， e_x 的相邻边集合 E_x 就相应变成了关于 v'_x 的相邻顶点集合 V'_x。具体做法是：把所有边 e_x 和邻居边集合 E_x 进行符号替换。

3) 根据图 G' 中的顶点 v'_x 和相邻顶点集合 V'_x，构造图 $G'=(V',E')$。具体做法是：画出所有顶点 v'_x，

然后把 v'_x 和 V'_x 里的所有顶点用线条连接起来。其中，顶点 v'_x 和相邻顶点集合 V'_x 的数学化表示为：

$$\begin{cases} v'_x \in V' \\ V'_x = \left\{ u' \middle| (u', v'_x) \in E' \right\} \end{cases} \tag{3}$$

我们应用以上方法对图进行变换，变换后的图如图所示。图 1(a)中 H 和 I 之间的 3 条边 a_1，a_2 和 a_3，经过变换后变成了图 1(b)中的 3 个顶点 a_1，a_2 和 a_3，b 和 c 之间的两条边 b_1 和 b_2 变成了图 1(b)中的 2 个顶点 b_1 和 b_2。

在图 1(a)中，我们用 7 种颜色来代表不同的加密应用。从图中可以看出，一种颜色对应了图中的多条边，这些边表示具有不同的 IP 对的同一种应用类型。在图 1(b)中，同样有 7 种颜色，图 1(a)中边的颜色对应图 1(b)中顶点的颜色。一种应用对应图 1(b)中多个顶点，在这些顶点中，有些是因为具有共同的服务器 IP 而连接在一起，而有些则不是(因为一种应用的服务器 IP 地址可能不唯一)。

2.3.2. 图嵌入过程

在前面的工作中，我们对图进行了变换，变换后的图中的顶点表示应用流，边表示应用流之间的连接关系。我们使用图嵌入技术[18] [19] [20]为这些顶点学习到低维的嵌入向量，如图 2 所示。具体过程是：首先通过随机游走算法遍历图中的顶点，得到顶点序列，这个顶点序列包含了当前顶点和相邻顶点的结构信息；然后使用自然语言处理中的词嵌入算法，为这些顶点序列生成词向量，也就是低维的嵌入向量。

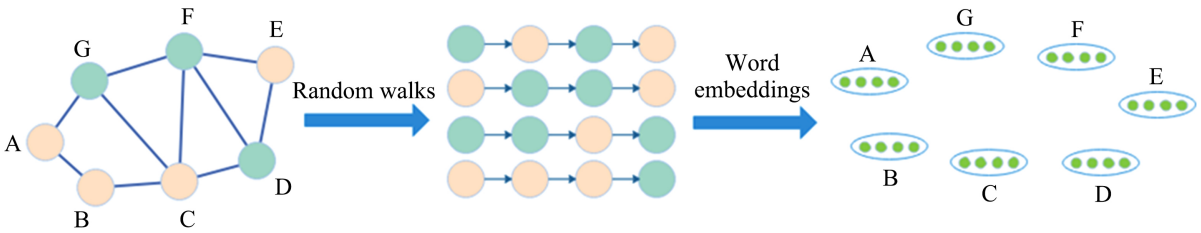


Figure 2. The diagram embedding process for an encryption app
图 2. 加密应用的图嵌入过程

随机游走是一个深度优先的图遍历算法。它的大致过程是这样的，给定访问的起始节点和序列长度，沿着边从一个顶点到另一个顶点进行游走，在所有下一个节点中，随机并且以相同概率选择下一个节点，直到访问(可重复访问已经访问过的节点)的节点序列长度和给定的长度相等。

$$p\left(c_i = x \middle| c_{i-1} = v\right) = \begin{cases} \frac{\pi_{vx}}{z} & \text{若} (v, x) \in E \\ 0 & \text{其它情况} \end{cases} \tag{4}$$

其中，c_i 表示第 i 次游走，π_{vx} 表示节点之间未归一化的转移概率，Z 为归一化常数。

词嵌入[20] [21] [22] [23]是一种把维度为词表大小的高维向量空间嵌入到一个低维连续向量空间的自然语言处理技术。每个单词都由一个在实数域上固定长度的低维向量表示，此向量隐含着当前词语的特征。Word2vec 词嵌入模型是一种无监督的浅层神经网络，它包含两个模型，skip-gram 和 CBOW (continuous bag of words)。在这里，图中节点可以看做是一个单词，随机游走的节点序列所组成的句子就是单词所在的上下文。不同的是，CBOW 通过上下文来预测目标单词，skip-gram 则通过目标单词预测上下文。它们都通过优化目标函数使概率最大化，得到的模型参数矩阵作为单词的词向量。我们使用 CBOW 来为加密应用流生成词嵌入向量。

给定一个句子$\left[w(1),w(2),\cdots,w(T)\right]$，其中 T 为句子的长度。CBOW 通过最大化目标函数 L，得到使目标函数最大化的权重参数矩阵。

$$L=\frac{1}{T}\sum_{t=1}^{T}\log P\left(w(t)\middle|w(t-k):w(t+k)\right) \tag{5}$$

$$P\left(w(t)\middle|w(t-k):w(t+k)\right)=\frac{\exp\left(h^{T}v(t)\right)}{\sum_{i=1}^{N}\exp\left(h^{T}v(i)\right)} \tag{6}$$

$$h=\frac{1}{2k}\sum_{-k\le j\le k,\,j\ne 0}u(t+j) \tag{7}$$

其中 k 为上下文窗口大小，N 为词典大小。$w(t)$ 为句子里的一个单词，$u(t)$ 为该单词的词向量，对应输入层到隐藏层的权重参数矩阵 U，$v(t)$ 为该单词对应隐藏层到输出层矩阵 V 的权重向量。

3. 多模态深度学习模型

我们的分类模型如图 3 所示。首先，我们把这两种特征(嵌入向量特征和包序列特征)合并成为加密应用的新特征；然后，我们使用多模态深度学习分类模型来同时学习这两种特征，使用卷积神经网络(CNN)处理图嵌入特征，使用门控循环单元(GRU)来处理包序列特征；最后，把这两种网络的输出合并输入到一个全连接层，再通过 softmax 函数输出一个概率向量。

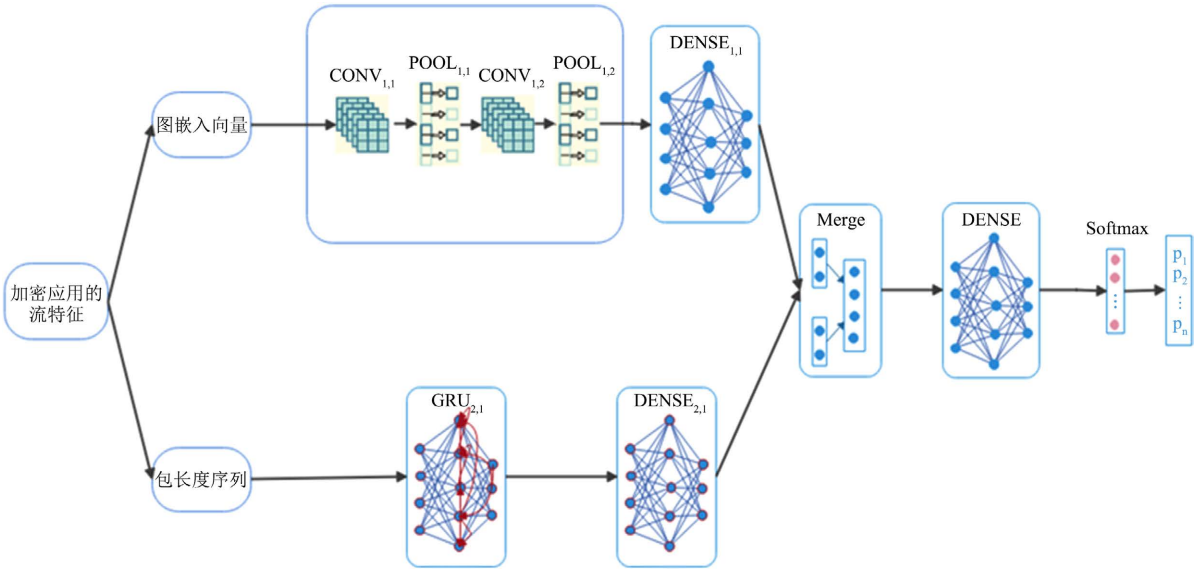


Figure 3. Multimodal deep learning classification algorithm
图 3. 多模态深度学习分类算法

4. 实验

4.1. 数据集

我们的流量数据来源于某大学网络的真实环境，这个数据集采集了 2021 年 8 月中 10 日到 17 日，总共 7 天的流量痕迹。为了能够更好地构建流量图，我们不能只收集一台主机网卡的流量，而是要收集不同主机的流量，并且这些不同主机网卡的流量也是在同一天产生的。我们对提取到的原始流量进行过滤，

只保留 SSL/TLS 协议的流量，大概总共提取了 89,375 条流(648,780 个包)。我们选取了诸如搜索，邮件，聊天，社交，购物，在线支付等应用服务的加密流量。表 1 列出了这些应用的名称以及它们的统计信息。

Table 1. Encrypts the app's flow and packet information
表 1. 加密应用的流和数据包信息

编号	应用	流	数据包
1	Alipay	7384	56,856
2	Baidu	15,622	141,258
3	Github	4887	49,654
4	Gmail	9206	45,109
5	JD	5601	45,702
6	LinkedIn	6198	60,125
7	QQ	10,367	72,569
8	Sogou	8910	53,463
9	Taobao	3329	37,941
10	Weibo	9112	46,471
11	Youdao	2718	10,817
12	Zhihu	6041	28,815
总和		89,375	648,780

4.2. 实验设置

除了一些众所周知的英文缩写，如 IP、CPU、FDA，所有的英文缩写在文中第一次出现时都应该给出其全称。文章标题中尽量避免使用生僻的英文缩写。

4.2.1. 评价标准

对于一条应用流，需要一个评价标准来判断它能够被正确分类以及错误分类的情况，因此我们使用两个有意义的指标来评估分类方法的性能：真阳性率和假阳性率(分别表示为 TPR 和 FPR)。另外，我们还使用 $\mathrm{TPR}_{\mathrm{AVE}}$ (所有正确分类流在所有流中的比率)，$\mathrm{FPR}_{\mathrm{AVE}}$ (所有错误分类流在所有流中的比率)和 FTF (TPR 和 FPR 的分数组合)来衡量模型的整体表现[24] [25] [26]。$\mathrm{TPR}_{\mathrm{AVE}}$ 和 $\mathrm{FPR}_{\mathrm{AVE}}$ 分别是所有应用 TPR 和 FPR 的平均值，这个平均值不是简单的算术平均，还考虑了这些应用流在所有流中的占比。FTF 这个指标是用来评价分类整体效果的。其中，TPR 越大，FPR 越小，这个值越大，表明分类效果越好。这 3 个指标定义如下：

$$\mathrm{TPR}_{\mathrm{AVE}} = \frac{1}{N}\sum_{i=0}^{C}\mathrm{TPR}_i \times N_i \tag{8}$$

$$\mathrm{FPR}_{\mathrm{AVE}} = \frac{1}{N}\sum_{i=0}^{C}\mathrm{FPR}_i \times N_i \tag{9}$$

$$\mathrm{FTF}=\frac{1}{N}\sum_{i=0}^{C}\frac{\mathrm{TPR}_i}{1+\mathrm{FPR}_i}\times N_i \tag{10}$$

其中 N_i 为第 i 个应用流的数目，C 表示应用数量，N 为总流量数，TPR_i 和 FPR_i 分别为第 i 个应用的评价标准。

4.2.2. 对比方法

为了更好地评估本文所提出方法——基于图嵌入和多模态深度学习的分类算法(GeMDL)的优劣，我们选取了 4 种当前表现比较好的加密流量分类算法，进行对比实验。这 4 种算法如下：

一阶马尔可夫链指纹模型(FOM) [24]：FOM 是最早的基于序列的加密流量识别方法，使用消息类型序列构建应用的一阶马尔可夫指纹。

二阶马尔可夫链指纹模型(SOB) [25] [26]：SOB 是具有应用属性的二阶马尔可夫指纹的代表，比 FOM 具有更高的准确率。

基于多属性的马尔可夫概率指纹(MaMPF) [27]：MaMPF [11]以消息类型的输出概率和长度块马尔可夫模型作为特征，使用随机森林分类器对加密流量进行分类。

流序列网络(FS-Net) [28]：FS-Net 是基于深度学习的加密流量分类模型的代表，它的优点依然是挖掘了流的特征序列。

4.3. 对比实验

通过对比实验来评估我们的模型，实验结果如表 2 所示。

Table 2. Compare experimental results
表 2. 对比实验结果

应用名称	FOM		SOB		MaMPF		FS-Net		GeMDL	
	TPR	FPR	TPR	FPR	TPR	FPR	TPR	FPR	TPR	FPR
Alipay	0.570	0.107	0.838	0.165	0.874	0.113	0.935	0.012	0.963	0.011
Baidu	0.633	0.260	0.862	0.160	0.817	0.074	0.863	0.143	0.937	0.091
Github	0.804	0.126	0.777	0.291	0.832	0.235	0.909	0.022	0.980	0.084
Gmail	0.708	0.297	0.890	0.126	0.946	0.062	0.764	0.058	0.941	0.022
JD	0.633	0.182	0.736	0.068	0.783	0.248	0.976	0.165	0.948	0.041
LinkedIn	0.763	0.466	0.685	0.314	0.699	0.122	0.931	0.011	0.961	0.037
QQ	0.579	0.378	0.521	0.373	0.772	0.058	0.864	0.216	0.936	0.009
Sogou	0.438	0.212	0.674	0.237	0.915	0.122	0.917	0.032	0.944	0.011
Taobao	0.628	0.304	0.857	0.162	0.814	0.231	0.772	0.168	0.956	0.083
Weibo	0.645	0.332	0.726	0.181	0.728	0.123	0.782	0.272	0.975	0.039
Youdao	0.594	0.255	0.801	0.156	0.842	0.207	0.834	0.101	0.928	0.045
Zhihu	0.375	0.431	0.616	0.203	0.771	0.352	0.897	0.217	0.861	0.114
平均	0.611	0.283	0.747	0.206	0.819	0.136	0.868	0.124	0.944	0.047
FTF	0.479		0.627		0.725		0.779		0.903	

从实验结果不难看出，从表中我们可以得出如下结论：

1) GeMDL 的表现要优于其他所有方法。我们在数据集上应用了 FOM、SOB、MaMPF、FS-Net 和 GeMDL，结果如表 2 所示。GeMDL 在 TPR_{AVE} (0.944)、FPR_{AVE} (0.047)和 FTF (0.903)中表现最好。

2) 相比于马尔可夫模型(FOM、SOB)，GeMDL 可以更好地对流序列进行建模。传统的基于马尔科夫的方法只能捕获一个流中相邻数据包的一到两个顺序信息，而 GeMDL 使用 GRU 网络对序列进行建模，具有保留整个流的上下文信息的优点。使用图嵌入方法来为主机的加密应用生成嵌入向量，仅仅只用到了加密的主机应用的连接关系，它的分类表现已经比 FOM 和 SOB 要好。这说明使用图嵌入技术为主机应用生成的词向量特征是

3) GeMDL 取得了比分段模型 MaMPF 和 FSNet 更好的性能。FSNet 仅将数据包长度序列作为其输入，MaMPF 结合了数据包长度序列和消息类型序列来对加密流量进行分类。然而，MaMPF 是一个分段模型，分类器无法指导从马尔可夫模型构建的特征。相反，FS-Net 受益于端到端的模型，可以弥补这个缺点，特征学习可以由分类任务和重建机制来指导。从而 FS-Net 比 MaMPF 表现要好。由于我们的 GeMDL 不光使用数据包长度序列，同时还结合了应用本身的图嵌入特征，这个图嵌入特征反映了一个主机的不同应用之间的关系。因此，GeMDL 的表现最好，并且还具有一定的抗干扰能力。

总之，如果只是考虑加密应用的流特征序列，它的分类性能有限。当攻击者对流量进行伪装，或者使流量特征模糊化，那么基于统计特征的流分类方法的准确率就会降低。然而，使用图嵌入特征的加密流量分类方法，攻击者是没办法修改其软件所产生流量的空间分布特性的。因此，基于图嵌入和多模态深度学习加密流量分类，不仅具有优秀的分类性能，还具有对网络攻击、对抗机器学习的抗干扰能力。

5. 总结

在本文中，我们设计了一种基于图嵌入和多模态深度学习的模型来对加密流量中的应用进行分类。该模型利用了加密应用的空间分布特征和序列特征，空间分布特征刻画了用户更偏好哪些应用，序列特征是加密应用的流统计特征。由于多模态深度学习的模型结合了这两种类型的特征进行训练，从而提高了分类性能。为了评估该模型的有效性，我们收集了真实网络环境的流量数据进行实验，对一些常用的应用进行分类。实验结果表明，本文所提出的方法在解决加密流量分类问题上具有出色的表现，并且优于最先进的方法。

参考文献

[1] Lin, C.H. and Lai, Y.Y. (2004) A Fingerprint-Based User Authentication Scheme for Multimedia Systems. *Proceedings of the* 2004 *IEEE International Conference on Multimedia & Expo* (*ICME* 2004), Taipei, 27-30 June 2004, 935-938.

[2] Internet Assigned Numbers Authority (2010) Port Numbers. http://www.iana.org/assignments/port-numbers

[3] Karagiannis, T., Broido, A., Faloutsos, M. and Claffy, K.C. (2004) Transport Layer Identification of P2P Traffic. *IMC*'04: *Proceedings of the* 4*th ACM SIGCOMM Conference on Internet Measurement*, 121-134. https://doi.org/10.1145/1028788.1028804

[4] Moore, A. and Papagiannaki, K. (2005) Toward the Accurate Identification of Network Applications. *Passive and Active Measurement Workshop* (*PAM* 2005), Boston, 31 March-1 April 2005, 41-54. https://doi.org/10.1007/978-3-540-31966-5_4

[5] Madhukar, A. and Williamson, C. (2006) A Longitudinal Study of P2P Traffic Classification. 14*th IEEE International Symposium on Modeling Analysis, and Simulation of Computer and Telecommunication Systems*, Monterey, 11-14 September 2006, 179-188.

[6] Knuth, D.E., Morris, J.H. and Pratt, V.R. (1977) Fast Pattern Matching in Strings. *SIAM Journal on Computing*, **6**, 323-350. https://doi.org/10.1137/0206024

[7] Boyer, R.S. and Moore, J.S. (1977) A Fast String Searching Algorithm. *Communications of the ACM*, **20**, 762-772.

https://doi.org/10.1145/359842.359859

[8] Aho, A.V. and Corasick, M.J. (1975) Efficient String Matching: An Aid to Bibliographic Search. *Communications of the ACM*, **18**, 333-340. https://doi.org/10.1145/360825.360855

[9] Cohn, D.A., Ghahramani, Z. and Jordan, M.I. (1996) Active Learning with Statistical Models. *Journal of Artificial Intelligence Research*, **4**, 129-145. https://doi.org/10.1613/jair.295

[10] Nguyen, T. and Armitage, G. (2007) A Survey of Techniques for Internet Traffic Classification Using Machine Learning. *IEEE Communications Surveys & Tutorials*, **10**, 56-76. https://doi.org/10.1109/SURV.2008.080406

[11] Lang, T., Aritage, G., Branch, P., *et al*. (2004) A Synthetic Traffic Model for Quake3. *Proceedings of* 2004 *ACM SIGCHI International Conference on Advances in Computer Entertainment Technology*, Singapore, 3-5 June 2004, 233-238. https://doi.org/10.1145/1067343.1067373

[12] McGregor, A.J., Hall, M.A., Lorier, P. and Brunskill, J. (2004) Flow Clustering Using Machine Learning Techniques. In: Barakat, C. and Pratt, I., Eds., *Passive and Active Network Measurement* (*PAM* 2004). *Lecture Notes in Computer Science*, Springer, Berlin, 205-214. https://doi.org/10.1007/978-3-540-24668-8_21

[13] Taylor, V.F., Spolaor, R., Conti, M. and Martinovic, I. (2016) AppScanner: Automatic Fingerprinting of Smartphone Apps from Encrypted Network Traffic. 2016 *IEEE European Symposium on Security and Privacy* (*EuroS&P*), Saarbruecken, 21-24 March 2016, 439-454. https://doi.org/10.1109/EuroSP.2016.40

[14] Conti, M., Mancini, L., Spolaor, R. and Verde, N. (2016) Analyzing Android Encrypted Network Traffic to Identify User Actions. *IEEE Transactions on Information Forensics and Security*, **11**, 114-125. https://doi.org/10.1109/TIFS.2015.2478741

[15] Anderson, B. and McGrew, D. (2016) Identifying Encrypted Malware Traffic with Contextual Flow Data. *ACM Workshop on Artificial Intelligence and Security*, Vienna, 28 October 2016, 35-46. https://doi.org/10.1145/2996758.2996768

[16] Anderson, B., Paul, S. and McGrew, D. (2016) Deciphering Malware's Use of TLS (without Decryption). arXiv:1607.01639.

[17] Anderson, B. and McGrew, D. (2017) Machine Learning for Encrypted Malware Traffic Classification: Accounting for Noisy Labels and Non-Stationarity. *Proceedings of the* 23*rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, Halifax, 13-17 August 2017, 1723-1732. https://doi.org/10.1145/3097983.3098163

[18] Liu, Z.M., Zheng, V.W., Zhao, Z., Zhu, F.W., Chang, K.C.C., Wu, M.H. and Ying, J. (2017) Semantic Proximity Search on Heterogeneous Graph by Proximity Embedding. *Proceedings of the Thirty-First AAAI Conference on Artificial Intelligence* (*AAAI*-17), **31**, 154-160.

[19] Scarselli, F., Gori, M., Tsoi, A.C., Hagenbuchner, M. and Monfardini, G. (2009) The Graph Neural Network Model. *IEEE Transactions on Neural Networks*, **20**, 61-80. https://doi.org/10.1109/TNN.2008.2005605

[20] Perozzi, B., Al-Rfou, R. and Skiena, S. (2014) DeepWalk: Online Learning of Social Representations. *Proceedings of the* 20*th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining* (*KDD* '14), New York, 24-27 August 2014, 701-710. https://doi.org/10.1145/2623330.2623732

[21] Grover, A. and Leskovec, J. (2016) Node2Vec: Scalable Feature Learning for Networks. *Proceedings of the* 22*nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining* (*KDD* '16), San Francisco, 13-17 August 2016, 855-864. https://doi.org/10.1145/2939672.2939754

[22] Ribeiro, L.F.R., Saverese, P.H.P. and Figueiredo, D.R. (2017) Struc2Vec: Learning Node Representations from Structural Identity. *Proceedings of the* 23*rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining* (*KDD* '17), Halifax, 13-17 August 2017, 385-394.

[23] Tang, J., Qu, M., Wang, M.Z., Zhang, M., Yan, J. and Mei, Q.Z. (2015) LINE: Large-Scale Information Network Embedding. *Proceedings of the* 24*th International Conference on World Wide Web* (*WWW*'15), Florence, 18-22 May 2015, 1067-1077. https://doi.org/10.1145/2736277.2741093

[24] Korczyński, M. and Duda, A. (2014) Markov Chain Fingerprinting to Classify Encrypted Traffic. *IEEE INFOCOM 2014—IEEE Conference on Computer Communications*, Toronto, 27 April-2 May 2014, 781-789. https://doi.org/10.1109/INFOCOM.2014.6848005

[25] Shen, M., Wei, M., Zhu, L. and Wang, M. (2017) Classification of Encrypted Traffic with Second-Order Markov Chains and Application Attribute Bigrams. *IEEE Transactions on Information Forensics and Security*, **12**, 1830-1843. https://doi.org/10.1109/TIFS.2017.2692682

[26] Shen, M., Wei, M., Zhu, L., Wang, M. and Li, F. (2016) Certificate-Aware Encrypted Traffic Classification Using Second-Order Markov Chain. 2016 *IEEE*/*ACM* 24*th International Symposium on Quality of Service* (*IWQoS*), Beijing, 20-21 June 2016, 1-10.

[27] Liu, C., Cao, Z.G., Xiong, G., Gou, G.P., Yiu, S.M. and He, L.T. (2018) Mampf: Encrypted Traffic Classification

Based on Multi-Attribute Markov Probability Fingerprints. 2018 *IEEE*/*ACM* 24*th International Symposium on Quality of Service* (*IWQoS*), Banff, 4-6 June 2018, 1-10.

[28] Chen, W., Jia, X., Chang, H.J., *et al*. (2021) Fs-Net: Fast Shape-Based Network for Category-Level 6D Object Pose Estimation with Decoupled Rotation Mechanism. *Proceedings of the IEEE*/*CVF Conference on Computer Vision and Pattern Recognition*, Nashville, 20-25 June 2021, 1581-1590. https://doi.org/10.1109/CVPR46437.2021.00163