

融合注意力机制的加密流量识别研究

郭子文 李启南

(兰州交通大学电子与信息工程学院 甘肃 兰州 730070)

摘要 针对加密流量识别技术在特征提取和选择效果不佳的问题,提出采用端到端结构思想,搭建融合注意力机制的 Attention-CNN 加密流量检测模型。利用卷积神经网络直接从流量数据中自动学习特征,使用经过 Softmax 激活函数的 attention 层动态捕捉到的信息对卷积层输出结果进行动态加权,通过全连接神经网络进行识别;实验采用 ISCXVPN2016 公共数据集,使用十折交叉验证的方法对模型进行验证;对注意力特征图与对应会话字节信息进行详细分析,找到具有特征信息的位置,解释关注的加密流量内容。实验结果表明,该方法相比现有方法有显著提升,同时对各类流量分类的评价指标取得较好效果。

关键词 加密流量 端到端 卷积神经网络 注意力机制 十折交叉验证

中图分类号 TP393.08

文献标志码 A

DOI:10.3969/j.issn.1000-386x.2025.12.047

ENCRYPTED TRAFFIC RECOGNITION WITH ATTENTION MECHANISM

Guo Ziwen Li Qinan

(School of Electronics and Information Engineering, Lanzhou Jiaotong University, Lanzhou 730070, Gansu, China)

Abstract For the problem of poor feature extraction and selection in encrypted traffic identification techniques, this paper proposes to adopt the idea of end-to-end structure, and to build an Attention-CNN encrypted traffic detection model with the attention mechanism. The convolutional neural networks were used to automatically learn features directly from traffic data. The information dynamically captured by the attention layer of the Softmax activation function was used to dynamically weighting the output of the convolutional layer. The fully connected neural network was used for recognition. The ISCXVPN2016 public dataset was used for experiments, and the model was verified by ten-fold cross validation method. The attention feature map and the corresponding session byte information were analyzed in detail. Locations with characteristic information were found and explained the content of encrypted traffic of attention. The experiment results show that this method has a significant improvement over the existing methods. At the same time the evaluation indexes for each type of traffic classification achieve better results.

Keywords Encrypted traffic End-to-end Convolutional neural network Attention mechanism Ten-fold cross validation

0 引言

据 2021 年思科安全网络分析数据表结果,如今大约 95% 的 Web 流量都经过加密,预计超过 70% 的攻击会使用加密^[1]。随着互联网和在线应用的迅速普及,网络流量分类越发重要。学者们已经做了大量关于流量分类的研究,产生许多不同的方法,并且都取得

一定的效果。互联网流量分类主要分为基于端口、基于有效载荷和统计的方法^[2]。基于端口的分类方法可以通过选择非标准端口来逃避检测,不同应用程序可以不依赖于固定端口,因此基于端口的方法不足以成为识别网络流量的特性^[3]。基于有效载荷的方法使用应用层协议的某些特征来对流量进行分类,但如今加密运用伪随机格式的数据包,使数据包有效载荷包含较少的恒定特征,对流量进行分类效果不佳^[4-5]。统

计方法通常使用数据包到达时间、大小及其他一些统计特性,需要依赖机器学习算法来进行流量分类,采用监督学习算法标记数据,建立合适的分类模型来进行训练^[6-7]。

文献[8]提出在规避流量五元组信息情况下,将加密流量划分为两个特征维度,通过报文负载和流指纹来描述网络流量的位置分布对流量进行识别。文献[9]提出一种基于神经自动编码器对流量的流进行无监督分类的方法,该方法利用流量中流的统计特性和聚类。结果表明,该算法检测七类不同的移动流量,平均准确率达到 80%,平均召回率达到 75%,但存在准确率较低的问题。在对流量数据的时序特征提取研究中,文献[10]使用自己采集的流量数据进行实验,对每条加密流量保留连接前 50 个包,提出使用 LSTM(长短记忆网络)提取流量包的时序特征,证明 LSTM 方法对网络流量数据的分类能力优于支持向量机、决策树等传统机器学习算法,但该方法问题在于一旦时间跨度大,就会存在梯度爆炸而导致准确率较低。文献[11]利用卷积自编码的无监督特征提取隐含层特征,将流量数据处理成 39×39 的图片搭建卷积神经网络模型对加密流量进行二分类和六分类的实验,二分类识别是否使用 VPN 的两类流量,六分类识别使用 VPN 加密的六类应用程序流量,六类识别中准确率达到 92.92%,但对常规加密和 VPN 加密的多类流量没有进行研究。文献[12]首次提出基于表征学习使用 2D_CNN(二维卷积神经网络)对加密流量进行分类,以流量数据为图像应用于软件流量分类,实验证明最佳流量表示类型是所有层的会话,但对流量识别的准确率并不高。文献[13]提出将 1D_CNN(一维卷积神经网络)应用于加密流量分类,流量本质为一维字节数据,使用 1D_CNN 刚好满足此条件,实验结果证明 1D_CNN 相比 2D_CNN 效果更好。文献[14]提出将流量拆分为流,在流中随机选择三个连续数据包,将预处理后的图像作为卷积神经网络输入,再将输出生成三个分组图像特征向量馈送到 LSTM 单元中,实验结果证明比 1D_CNN 模型准确率高,时间花费少。针对加密流量识别技术在特征提取和选择效果不佳的问题,本文提出采用端到端结构思想,搭建融合注意力机制的 Attention-CNN 加密流量检测模型。本文的主要工作如下:

(1) 本文提出搭建融合注意力机制和卷积神经网络的端到端方法来对加密流量的特征进行进一步压缩和提取。使用经过激活函数的注意力层的特征层进行动态赋权,最后对中间注意力图层进行可视化。

(2) 采用 ISCXVPN2016 公共数据集^[15]进行实验,实验结果表明,相比于 2D_CNN^[12]、1D_CNN^[13] 和 CNN + LSTM^[14],本文方法对流量识别取得更好的识别效果,对单类流量识别的各项评价指标也达到较好的效果。

(3) 字节分析。通过对注意力层得到的注意力图与原始流量数据的二进制文件进行映射,找到注意力图关注到的位置,并对关注到的流量信息进行详细分析。

1 方 法

1.1 数据集预处理

本次实验选取的数据集是加拿大网络安全研究所(Canadian Institute for Cybersecurity)纽布伦斯威克大学(University of New Brunswick)官方网页发布的 VPN-nonVPN dataset(ISCXVPN2016)公开数据集。在加密流量研究领域,该数据集被广泛使用。该数据集总共有 14 个流量类别:VOIP、VPN-VOIP、P2P、VPN-P2P 等。在本次实验中只使用十二种流量,比如浏览器类(browser)流量属于多个类别,不能完全归为一类,所以本次实验中不考虑此类流量。举例只挑选 1~5 个文件,使用的十二类流量如表 1 所示。

表 1 数据集部分文件名称

类型	应用名称
Chat	aim, facebook, icq, skype, hangout
Email	email, gmail
File	skype
P2p	torrent
Streaming	netflix, spotify, vimeo, youtube, youtubeHTML
Voip	hangouts_audio
Vpn_Chat	aim, facebook, hangout, icq, skype
Vpn_Email	email
Vpn_File	ftp, sftp, skype
Vpn_P2p	bittorrent
Vpn_Streaming	netflix, spotify, vimeo, youtube
Vpn_Voip	facebook, hangout, skype, voipbuster

数据集 ISCXVPN2016 每类流量对应在一个文件夹,使用以下方法对数据进行预处理:

(1) 格式统一。文件中有 AIMchat1.pcapng 和

Facebookchat2.pcapng 等以.pcapng 文件存在的格式。通过使用 wireshark 软件对.pcapng 文件格式进行转换为 pcap,处理之后的所有文件统一为.pcap 格式。

(2) 流量切分。对流量进行切分就是拆包操作。文献[11]中实验证明对流切分比会话切分的效果更差,会话双向流比单向流保留更多有效信息。本文将.pcap 文件进行会话层面的拆分。

(3) 流量清洗。流量文件中包含数据链路层中的 MAC 硬件地址和 IP 地址,均不是区分流量的有效信息,还会影响到流量识别效果,故对这些信息的数据包字符串进行删除。

(4) 流量采样。对切分好的流量文件.pcap 包进行预处理,对流量数据进行下采样操作。为使样本更加平均,本文对每类流量随机采样 6 000,数量不足的流量文件以原有数量进行实验。

(5) 统一尺寸,生成图片,转换为 IDX 格式。为了统一图像大小尺寸,在生成图片之前先将尺寸统一为 784 比特,其中切分后的数据大于 784 比特则进行下采样操作进行截断,若不足 784 比特的文件则用 0x00 对字节进行填充。将 784 比特的流量数据按照二进制转换为 28 × 28 的图片,字节对应灰度像素值,如 0xff 是白色。将生成后训练、测试的图片和标签四个文件夹都使用 IDX 格式进行打包。

如图 1 所示,两幅图选取的都是 Streaming(流)类应用程序 Netflix 的流量,图 1(a)是常规加密 Streaming 流量,图 1(b)是 VPN 加密 Streaming 流量。从图 1(a)和图 1(b)中可以看出虽是同一个程序,但不同加密方式流量的图片纹理不同,其对应流量字节信息均不同。如图 2 所示,随机选取两幅图片均来自 Vpn_Chat(聊天)类,可以看出相同加密方式图片很多部分都有相似纹理。图 3 是图 1(a)常规 Streaming 流量对应内容,共有九条传输控制协议(Transmission Control Protocol, TCP)会话。从图 4 中可以看到 VPN 加密 Streaming 流量采用了安全传输层协议(Transport Layer Security, TLS),TLS 协议属于加密协议,在网络数据传输中起到保护隐私和数据的完整性的作用。图 2(a)对应会话内容为图 5,图 2(b)对应会话内容为图 6,从图 5 和图 6 中可以看到同类加密方式流量对应会话内容中的源地址、目的地址、协议、长度等特征一致,再将对应每条会话仔细分析发现只有目的端口号不一致,使用加密协议 TLS 会话中产生的随机值不一致等细微区别。从流量可视化的角度来分析,使用流量图片分类的方法应取得较好结果。

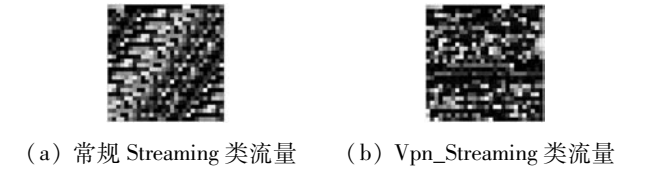


图 1 不同类流量

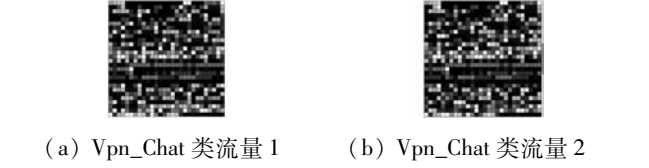


图 2 同类型流量

No	Time	Source	Destination	Protocol	Length	Info
1	0.000000	131.202.240.150	23.195.206.176	TCP	66	6639686 → 443[ACK]
2	0.031880	23.195.206.176	131.202.240.150	TCP	66	[TCP ACK'd unseen
3	45.056010	131.202.240.150	23.195.206.176	TCP	66	[TCP Dup ACK 1#1]
4	45.085759	23.195.206.176	131.202.240.150	TCP	66	[TCP Dup ACK 2#1]
5	90.112014	131.202.240.150	23.195.206.176	TCP	66	[TCP Dup ACK 1#2]
6	90.143798	23.195.206.176	131.202.240.150	TCP	66	[TCP Dup ACK 2#2]
7	135.168075	131.202.240.150	23.195.206.176	TCP	66	[TCP Dup ACK 1#3]
8	135.198505	23.195.206.176	131.202.240.150	TCP	66	[TCP Dup ACK 2#3]
9	180.224015	131.202.240.150	23.195.206.176	TCP	66	[TCP Dup ACK 1#4]

图 3 常规 Streaming 类对应的流量信息

No	Time	Source	Destination	Protocol	Length	Info
1	0.000000	10.8.8.138	107.22.243.234	TCP	60	35936 → 443
2	0.219757	107.22.243.234	10.8.8.138	TCP	60	443 → 35936
3	0.219783	10.8.8.138	107.22.243.234	TCP	52	35936 → 443
4	0.219970	10.8.8.138	107.22.243.234	TLSv1	255	Client Hello
5	0.441004	107.22.243.234	10.8.8.138	TCP	52	443 → 35936

图 4 Vpn_Streaming 类对应的流量信息

No	Time	Source	Destination	Protocol	Length	Info
1	0.000000	10.8.8.178	23.192.162.186	TCP	60	46884 → 443[SYN] - Tsval=3612968 ..
2	0.130875	23.192.162.186	10.8.8.178	TCP	60	443 → 46884[SYN,ACK] Seq=0 Ack=1 ..
3	0.130915	10.8.8.178	23.192.162.186	TCP	52	46884 → 443[ACK] - Tsval=3613000 ..
4	0.133543	10.8.8.178	23.192.162.186	TLSv1	300	Client Hello
5	0.264682	23.192.162.186	10.8.8.178	TCP	52	443 → 46884[ACK]-Tsval=2285785730..

图 5 Vpn_Chat 类流量 1 对应的流量信息

No	Time	Source	Destination	Protocol	Length	Info
1	0.000000	10.8.8.178	23.192.162.186	TCP	60	46887 → 443[SYN] - Tsval=3613074 ..
2	0.129525	23.192.162.186	10.8.8.178	TCP	60	443 → 46884[SYN,ACK] Seq=0 Ack=1 ..
3	0.129579	10.8.8.178	23.192.162.186	TCP	52	46884 → 443[ACK] - Tsval=3613107 ..
4	0.129859	10.8.8.178	23.192.162.186	TLSv1	300	Client Hello
5	0.259655	23.192.162.186	10.8.8.178	TCP	52	443 → 46884[ACK]-Tsval=2285786152..

图 6 Vpn_Chat 类流量 2 对应的流量信息

1.2 模型

目前主流的流量分类方法分为基于机器学习和基于深度学习的流量分类。机器学习流量分类方法的弊端是需要专家人工提前设定提取模式和特征来对流量进行分类^[16],需要消耗大量的人力与财力,且容易出错。相比之下,基于深度学习的方法对流量数据的特征提取和分类阶段集成到一个系统当中,中间的学习过程不需要人为干预,有效地避免机器学习的弊端,从而在流量分类研究中取得一定的优势。本文整体采用端到端方法来对加密流量进行识别,流程如图 7 所示,首先将原始流量进行数据预处理,然后将处理后的数据输入到加密流量识别的分类模型中进行训练,对流量进行特征提取,最后保存模型进行验证后将识别结果输出。

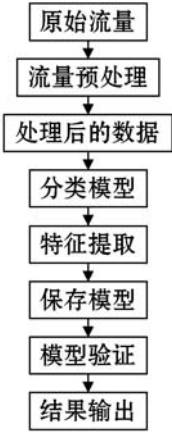


图7 端到端结构

卷积神经网络是深度学习领域中代表算法之一,在近年来用于处理图像识别的效果较好。本文提出融合注意力机制和卷积神经网络的端到端加密流量分类方法,参考经典卷积神经网络模型 Lenet5 的卷积层,还引入池化层、平滑层、全连接层,同时引入注意力机制能够将网络中间层的特征赋予相应的权重并进行可视化。本文搭建如图8所示模型进行训练与测试。

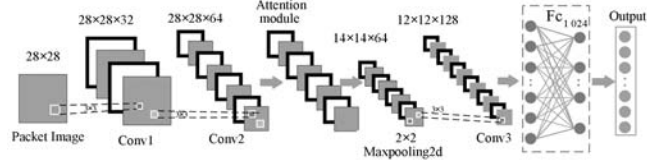


图8 加密流量分类模型结构

模型的输入经过流量会话数据预处理后的图片,首先增加 32 个神经元通道的卷积层,卷积核大小为 3×3 ,填充为 same;然后再增加 64 个神经元的卷积层,卷积核大小为 3×3 ,填充为 same。卷积式子如下:

$$Z^P = \sum_{(d=1)}^D W^{(p,d)} \otimes X^d + b^P \quad (1)$$

$$Y^P = f(Z^P) \quad (2)$$

式中: D 代表有 D 个输入特征映射, P 代表有 P 个输出特征映射, d 指当前为第 d 个输入特征映射, $d = [1, D]$, $\otimes$ 表示卷积运算, Z^P 表示用卷积核 $W^{(p,d)}$ 对第 d 个输入特征映射 X^d 进行卷积, D 个卷积结果进行相加,并加上标量偏置 b^P 。最后经过非线性函数输出得到特征映射 Y^P 。

在第二层卷积之后增加注意力层,注意层得到的权重与上一层得到的特征相乘。本文所使用的注意力机制的原理如图9所示,首先从模型第二层卷积层的输出得到 $a_{i,j}$ ($0 \leq i \leq W, 0 \leq j \leq H$),然后将得到的 $a_{i,j}$ 传输到注意力层,计算出图像的注意力权重 $\hat{a}$,然后再传输到模型的下一层^[17]。式子如下:

$$X = \text{softmax} \odot Fc_{(w_m, b_m)} \odot \text{ReLU} \odot Fc_{(w_a, b_a)} \quad (3)$$

式中:计算图像的注意力权重 $\hat{a}$,注意力机制计算 $a_{i,j}$

的每个区域的注意力权重为 $x_{i,j}$, $a_{i,j}$ 表示从上一层卷积层输出得到的特征图, $X = x_{i,j}$ 表示对每个输入特征强调的总和, $Fc_{(w_a, b_a)}$ 表示全连接输入向量, $\odot$ 表示向量元素相乘。然后再计算注意力权重的平均值,如式(4)所示。

$$\hat{a} = \sum_{i=1}^W \sum_{j=1}^H x_{(i,j)} a_{(i,j)} \quad (4)$$

式中: $a_{i,j}$ 表示从上一层卷积层得到的特征图的某个区域, $x_{i,j}$ 表示得到特征图某个区域的注意力权重。

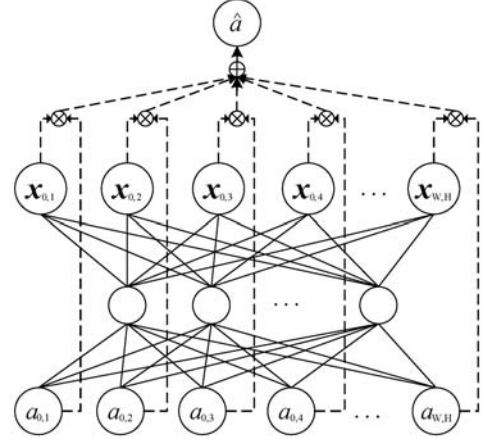


图9 注意力机制

神经网络学习随着迭代次数增多,学习到的特征权重参数就越大,在注意力层后增加 64 个神经元通道的池化层,在特征学习时采用 2×2 大小的核对特征图片进行压缩,填充为 same。本次实验采用最大池化,对某区域 $R_{(m,n)}^d$ 内选择神经元的最大值作为其区域的表示,如式(5)所示。

$$y_{(m,n)}^d = \max_{i \in R_{(m,n)}^d} x_i \quad (5)$$

池化层之后为进一步提取流量特征信息,再增加 128 个神经元通道的卷积层,卷积核大小为 3×3 ,填充为 same。卷积层后增加平滑层,将得到的特征图全部拉成一维向量。增加包含 1 024 个神经单元的平滑层,激活函数为 ReLU。ReLU 函数是输入值 x 大于零时,输出 x 值;当 x 小于等于零时,则输出 0。如式(6)所示。

$$h(x) = \begin{cases} x & x > 0 \\ 0 & x \leq 0 \end{cases} \quad (6)$$

最后增加包含 12 个神经元的输出层全连接层,输出函数为 softmax,最后输出结果类别标签。softmax 的输出为 0 到 1 间的实数,输出值总和为 1。假设有 n 个神经元,计算第 k 个神经元的输出为 y_k 。如式(7)所示。

$$y_k(a) = \frac{\exp(a_k)}{\sum_{i=1}^n \exp(a_i)} \quad (7)$$

式中:exp 代表计算输入值的指数函数,分子是输入信号 a_k 的指数函数,分母是指所有输入信号指数函数的和.softmax 输出总值为 1,如式(8)所示。

$$\sum_{j=1}^i \sigma_j(z) = 1 \tag{8}$$

2 实验环境和评估指标

2.1 实验环境和参数

本文实验操作系统为 Windows 10 专业版,开发语言环境为 Python 3.8.13,深度学习平台 Keras 2.3.1,后台框架为 TensorFlow-GPU 2.4.0。实验使用 GPU 进行加速,显卡为 Nvidia GeForce GTX1650 Ti。

神经网络最后一层通过 softmax 函数获得输出,模型预测的类别概率输出与真实类别的独热编码形式进行交叉损失函数的计算。交叉熵损失函数如下:

$$L = \frac{1}{N} \sum_i L_i = \frac{1}{N} \sum_i \sum_{c=1}^M y_{ic} \log(p_{ic}) \tag{9}$$

式中: M 代表流量类别的数量为 12; y_{ic} 是符号函数,若预测样本 i 的真实类别为 c ,则值取 1,不是则取 0; p_{ic} 是预测样本 i 属于类别 c 的预测概率。

在模型进行训练时,不同的优化函数就会得到不同的结果。本文选取 SGD、Adagrad、RMSprop、Adam^[18] 四种优化器对损失率影响进行对比,图 10 是对比结果,可以看出 SGD 和 Adagrad 收敛效果较差,Adam 和 RMSprop 可以快速收敛,但 Adam 的效果最好,损失率降到最低。故本文使用 Adam 作为优化器。

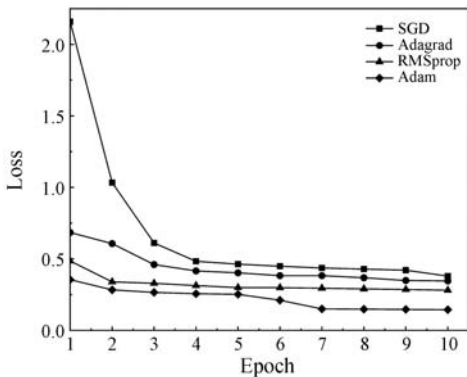


图 10 不同优化函数对损失率影响

训练样本数据如图 11 所示,将训练集和测试集分为 9:1。此次实验采用的测试集也是验证集样本。可以看出,VPN 协议加密的 Streaming、P2P、Email、File_Transfer 四类流量样本还不足 1 000,达到 6 000 样本的只有五类流量。部分类别的数据样本较少导致数据集样本不均衡的问题,故本文采用十折交叉验证法,将训练数据集划分为 10 份,其中 9 份用于模型的训练,剩余的 1 份用于测试。重复此过程 10 次共得到 10 次

对模型性能的评价。最后,取 10 次结果的均值作为对算法精度的估计。

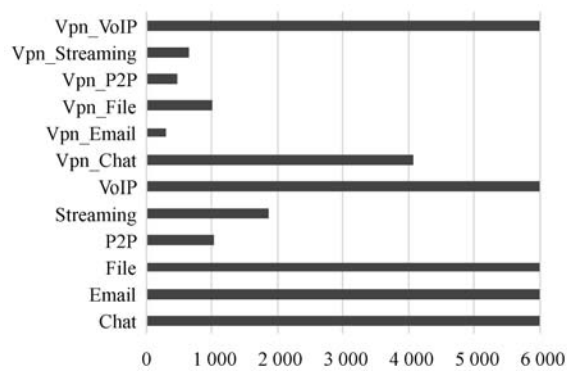


图 11 ISCXVPN2016 选取样本数据量

2.2 评估指标

本次实验是多分类实验,采用准确率(Accuracy)、精确率(Precision)、召回率(Recall),还有 F1 分数(F1-score)这四项标准指标作为评判本文模型对流量的分类效果。其中 TP 表示实际为 A 类,预测结果也是 A 类,即 A 类流量预测正确的结果;TN 表示实际为 B 类,预测结果也为 B 类,表示 12 类中其他 11 类流量实际和预测结果正确的概率;FN 表示实际为 A 类,但被误判预测为 B 类,即真实类别误判为其他 11 类的结果;FP 表示实际为其他 11 类的流量被误判为 A 类的结果。准确率、精确率、召回率、F1 分数四项评估指标式子如下:

$$A_{accuracy} = \frac{T_p + T_n}{T_p + T_n + F_p + F_n} \tag{10}$$

$$P_{recision} = \frac{T_p}{T_p + F_p} \tag{11}$$

$$R_{ecall} = \frac{T_p}{T_p + F_n} \tag{12}$$

$$F_{1-Score} = \frac{2 \times P_{recision} \times R_{ecall}}{P_{recision} + R_{ecall}} \tag{13}$$

3 实验结果与分析

3.1 分类模型对加密流量分类的结果

每类流量识别的精确率、召回率和 F1 分数如表 2 所示,各类流量的精确率均可以达到 0.91 以上。其中 Email 流量和 Vpn_P2P 流量的精确率达到 1,说明对这两类流量的识别效果好,File 流量较低只有 0.912。召回率均可以达到 0.91 以上,其中对 Voip 的识别效果较差只有 0.91,而对 P2P、Streaming、Vpn_Streaming 流量的识别率较高。最后,流量分类的 F1 分数都在 0.93 以上,对 File 和 Voip 流量识别分数较低,其他十类识别分数都很高。

表 2 本文方法对加密流量的各类流量分类结果

类型	Precision	Recall	F1-socre
Chat	0.992	0.998	0.995
Email	1.000	0.992	0.996
File	0.912	0.950	0.931
P2p	0.981	1.000	0.991
Streaming	0.984	1.000	0.992
Voip	0.955	0.910	0.932
Vpn_Chat	0.993	1.000	0.996
Vpn_Email	0.967	0.967	0.967
Vpn_File	0.990	0.971	0.980
Vpn_P2p	1.000	0.958	0.979
Vpn_Streaming	0.985	1.000	0.993
Vpn_Voip	0.998	0.997	0.998

图 12 是本文方法对十二类流量识别得到的混淆矩阵,矩阵是以图片幅数为单位的,纵坐标表示真实标签,横坐标表示预测标签。可以看出本文方法对十二类流量的识别误判很低,只有真实标签为 Voip 类流量 54 幅图片被误判为 File 类流量,以及 File 类流量中有 26 幅图片被误判为 Voip 类流量,与表 2 中的单分类识别率相对应。还有 Email 流量有五幅被预测误判为 Chat,像 Vpn_Chat、VPN_Streaming 这类流量误判会有 1 到 2 幅。

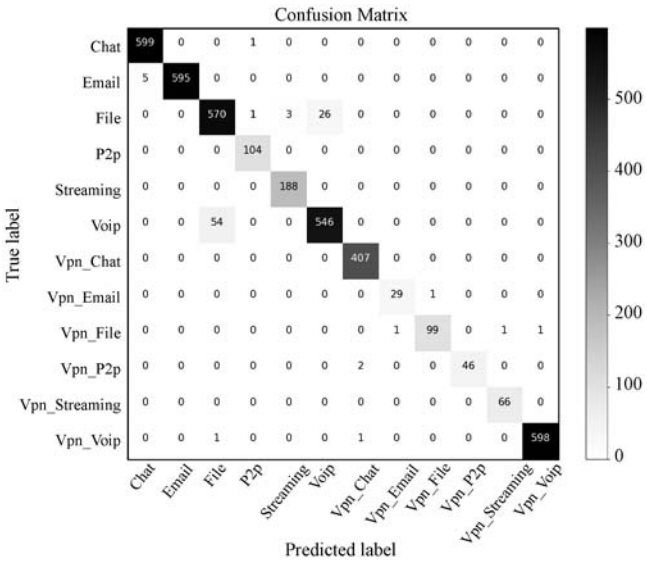


图 12 本文方法混淆矩阵

为验证本文方法的有效性和可靠性,本文对 2D_CNN、1D_CNN、CNN + LSTM、本文方法在 ISCX-VPN2016 数据集的实验结果进行对比^[19]结果如表 3 所示。实验结果表明,本文方法对加密流量分类的准确率达到 97.5%,相比 2D_CNN、1D_CNN 和 CNN + LSTM 分别提高 12.9 百分点、11.2 百分点和 6 百分

点。本文方法对加密流量分类的精确率、召回率、F1 分数都有提升,精确率相比三种方法分别提高了 10.4 百分点、7.8 百分点和 8.3 百分点,召回率分别提高 10.7 百分点、8.8 百分点和 8.1 百分点,F1 分数分别提高 11.4 百分点、8.6 百分点和 8.2 百分点,说明本文方法对流量分类达到很好的效果。

表 3 2D_CNN、1D_CNN、CNN + LSTM 和本文方法的比较结果(%)

分类方法	Accuracy	Precision	Recall	F1-score
2D_CNN	84.6	87.6	87.2	86.5
1D_CNN	86.3	90.2	89.1	89.3
CNN + LSTM	91.5	89.7	89.8	89.7
本文方法	97.5	98.0	97.9	97.9

为进一步验证本文方法对加密流量多分类任务的识别性能,还对比四种方法、十二类流量的精确率。从图 13 中可以发现,三种对比模型都有识别率较低类别,其中:2D_CNN 对 Chat 类流量的精确率只有 0.671,1D_CNN 对 Chat 类识别精确率为 0.655,而 CNN + LSTM 对 Voip 类流量的精确率为 0.744。本文方法除对 File 和 VPN_Email 的识别精确率低于 1D_CNN 外,其他十类流量的精确率均高于对比模型。

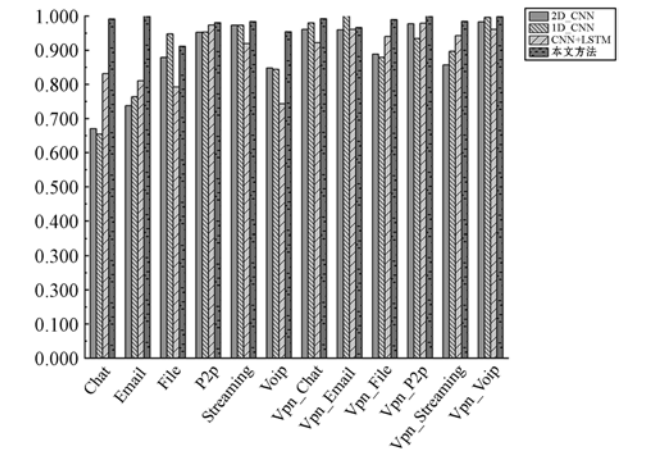


图 13 2D_CNN、1D_CNN、CNN + LSTM 和本文方法精确率比较

3.2 可视化分析

在可视化分析中,本文对流量会话字节内容做分析,随机选取 VPN_FTP 文件的一条流量。该流量文件通过 wireshark 流量分析工具,对应 5 条会话如图 14 所示。

No	Time	Source	Destination	Protocol	Length	Info
1	0.000000	10.8.8.138	204.236.238.164	TCP	60	32797→1742[SYN] Seq=0 Win=29200 ...
2	0.219066	204.236.238.164	10.8.8.138	TCP	60	1742→32797 [SYN,ACK] Seq=0 Ack=1 ...
3	0.219117	10.8.8.138	204.236.238.164	TCP	52	32797→1742[ACK] Seq=1 Ack=1 ...
4	0.219469	10.8.8.138	204.236.238.164	SSLv3	143	Client Hello
5	0.444923	204.236.238.164	10.8.8.138	TCP	52	443 → 46884[ACK] Seq=1 Ack=92 ...

图 14 .pcap 文件对应五条会话

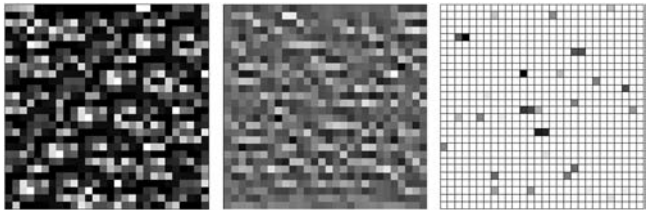
5 条会话对应字节如图 15 所示。图 15 第一个框第一个字节为 45,其转换为 16 进制后像素值为 69。图 15 中圈出的前三个和最后一个框均为 TCP 协议会

话,第四条圈出会话是对应 SSLv3 加密协议会话,图 15 中每条会话中黑色实线对应的是会话的 IPv4 中的信息,双实线和波浪线都是传输控制协议内容,其中波浪线是传输控制协议中的 TCP 报文头部选项字段(TCP Options)。第四条框中的虚线是传输层安全内容。图 15 中所有标灰部分为注意力可视化图层关注到的信息。原始流量图片是 28×28 的图片,为方便分析用画图工具将其放大,如图 16(a)所示。



图 15 原始流量会话对应字节

选取这幅流量图片做详细分析,在保存训练模型文件后对其进行验证,将训练好的模型的 attention(注意力)层抽取出来,作为新样本的 attention 抽取器。下面对样本的 attention map(注意力特征图)进行可视化。从图 16 中可以看出,图 16(b)卷积层 feature map(特征图)抽取的特征与图 16(a)存在明显差异。attention 层在卷积层之后,attention 层学习拟合到的特征存在差异。从图 16(b)卷积层 feature map 输出后的图片与图 16(c)的 attention map 输出的图片可以看出它们存在明显差异。



(a) 原始流量图像 (b) 卷积特征图 (c) 注意力特征图
图 16 特征图

图 15 中灰色部分为图 16(b)注意力特征图的关注点,对关注点对应的字节作详细分析,如 2 行 16 列关注到的值为 3c,对应字段十进制值为 60,是第一条 TCP 会话中 IPv4 信息的总长度 60;10 行 12 列关注像素点对应值为 8f,对应字段十进制值为 143,其关注的信息是第四条 SSL 加密协议会话中 IPv4 信息的总长度为 143,如图 17 所示;16 行第 3 列关注像素点值为 34,对应字段十进制为 52,关注的信息为第五条会话中的 IPv4 总长度为 52。除关注到会话长度的特征外,

其余关注到的信息还包括 10 行 17 列关注像素点值与 16 行 6 列关注像素点值一样均为 40,其对应信息是标记位(flags)40,以及 SSL 加密协议中握手过程中的扩展长度等信息。

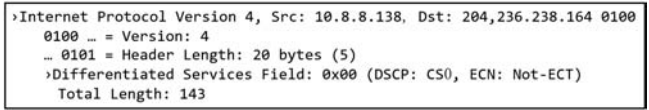


图 17 SSL 加密协议对应 IPv4 内容

4 结 语

针对加密流量识别技术在特征提取和选择效果不佳的问题,本文搭建融合注意力机制和卷积神经网络的模型,对加密流量进行分类,通过实验并与其他流量分类模型进行对比取得较好的效果,在此前的相关参考文献中,没有对流量数据以及对应的二进制文件映射关系进行分析,本文对注意力特征图与图中对应字节信息进行详细分析,找到具有特征信息的位置,解释加密流量对应关注内容。但也存在不足,在本次研究中是将一维流量字节序列进行数据处理转成灰度图片,在数据截取的过程中,对部分字节的截取操作丢失部分有效信息,今后的研究中可以对字节序列进行更深入的研究,搭建更优化的模型。

参 考 文 献

[1] Cisco. Cisco secure network analytics (formerly stealthwatch) datasheet[EB/OL]. [2021-08-15]. <https://www.cisco.com/c/en/us/products/collateral/security/datasheet-c78-739398.html>.
[2] 陈良臣,高曙,刘宝旭,等.网络加密流量识别研究进展及发展趋势[J].信息安全学报,2019(3):19-25.
[3] Rezaei S, Liu X. Deep learning for encrypted traffic classification: An overview[J]. IEEE Communications Magazine, 2019,57(5):76-81.
[4] Zhang J, Chen X, Xiang Y, et al. Robust network traffic classification[J]. IEEE Transactions on Networking, 2014, 23(4):1257-1270.
[5] Abbasi M, Shahraki A, Taherkordi A. Deep learning for network traffic monitoring and analysis(NTMA): A survey[J]. Computer Communications, 2021,170:19-41.
[6] Liu C, Cao Z G, Xiong G, et al. MaMFP: Encrypted traffic classification based on multi-attribute Markov probability fingerprints[C]//26th International Symposium on Quality of Service, 2018:1-10.
[7] 曾勇,吴正远,董丽华,等.加密流量中的恶意流量识别技术[J].西安电子科技大学学报,2021,48(3):170-187.

参 考 文 献

- [1] Hu Y, Sun Y, Wang Y, et al. An enhanced multi-stage semantic attack against industrial control systems[J]. IEEE Access, 2019, 7: 156871 – 156882.
- [2] 方栋梁, 刘圃卓, 秦川, 等. 工业控制系统协议安全综述[J]. 计算机研究与发展, 2022, 59(5): 978 – 993.
- [3] 王文博, 刘绚, 张博, 等. 基于协议特征的电力工控网络流量异常行为检测方法[J]. 电力系统自动化, 2023, 47(2): 137 – 145.
- [4] Lee R M, Assante M J, Conway T. German steel mill cyber attack[J]. Industrial Control Systems, 2014, 30(62): 1 – 15.
- [5] 陈思, 吴秋新, 张铭坤, 等. 基于边云协同的智能工控系统入侵检测技术[J]. 计算机应用与软件, 2020, 37(11): 280 – 285, 333.
- [6] 张志飞, 王露漫. 基于机器学习的网络入侵检测算法研究[J]. 计算机应用与软件, 2022, 39(10): 336 – 343.
- [7] Kakihata E M, Sapia H M, Oikawa R T, et al. Intrusion detection system based on flows using machine learning algorithms[J]. IEEE Latin America Transactions, 2017, 15(10): 1988 – 1993.
- [8] Abubakar A, Pranggono B. Machine learning based intrusion detection system for software defined networks[C]//2017 7th International Conference on Emerging Security Technologies (EST), 2017: 138 – 143.
- [9] Othman S M, Ba-Alwi F M, Alsohybe N T, et al. Intrusion detection model using machine learning algorithm on Big Data environment[J]. Journal of Big Data, 2018, 5(1): 34.
- [10] 顾兆军, 李冰, 刘涛. 基于 ELM-KNN 算法的网络入侵检测模型[J]. 计算机工程与设计, 2018, 39(8): 2412 – 2416, 2421.
- [11] 谭敏生, 杨帅创, 丁琳, 等. 结合随机森林的 PSO-CNN 入侵检测研究[J]. 计算机应用与软件, 2021, 38(12): 326 – 331.
- [12] Sun P, Liu P, Li Q, et al. DL-IDS: Extracting features using CNN-LSTM hybrid network for intrusion detection system[J]. Security and Communication Networks, 2020, 2020: 8890306.
- [13] Ling J, Zhu Z, Luo Y, et al. An intrusion detection method for industrial control systems based on bidirectional simple recurrent unit[J]. Computers & Electrical Engineering, 2021, 91: 107049.
- [14] Chawla N V, Bowyer K W, Hall L O, et al. SMOTE: Synthetic minority over-sampling technique[J]. Journal of Artificial Intelligence Research, 2002, 16: 321 – 357.
- [15] 朱继洪, 裴继红, 赵阳. 卷积神经网络(CNN)训练中卷积核初始化方法研究[J]. 信号处理, 2019, 35(4): 641 – 648.
- [16] 张梦楠, 李红娇. 基于 DCNN 和 SVC 的窃电检测[J]. 计算机仿真, 2022, 39(6): 92 – 97, 429.
- [17] Lei T, Zhang Y, Wang S I, et al. Simple recurrent units for highly parallelizable recurrence[C]//2018 Conference on Empirical Methods in Natural Language Processing, 2018: 4470 – 4481.
- [18] Morris T, Gao W. Industrial control system traffic data sets for intrusion detection research[C]//International Conference on Critical Infrastructure Protection, 2014: 65 – 78.
- ~~~~~
- (上接第 355 页)
- [8] 胡斌, 周志洪, 姚立红, 等. 报文负载与流指纹特征的恶意流量检测[J]. 计算机工程, 2020, 46(11): 157 – 163.
- [9] Höchst J, Baumgärtner L, Hollick M, et al. Unsupervised traffic flow classification using a neural autoencoder[C]//42nd Conference on Local Computer Networks, 2017: 523 – 526.
- [10] 邹源, 张甲, 江滨. 基于 LSTM 循环神经网络的恶意加密流量检测[J]. 计算机应用与软件, 2020, 37(2): 308 – 312.
- [11] Guo L, Wu Q, Liu S L, et al. Deep learning-based real-time VPN encrypted traffic identification methods[J]. Journal of Real-Time Image Processing, 2020, 17(1): 103 – 114.
- [12] Wang W, Zhu M, Zeng X W, et al. Malware traffic classification using convolutional neural network for representation learning[C]//International Conference on Information Networking, 2017: 712 – 717.
- [13] Wang W, Zhu M, Wang J L, et al. End-to-end encrypted traffic classification with one-dimensional convolution neural networks[C]//IEEE International Conference on Intelligence and Security Informatics, 2017: 43 – 48.
- [14] Zou Z, Ge J G, Zheng H B, et al. Encrypted traffic classification with a convolutional long short-term memory neural network[C]//20th International Conference on High Performance Computing and Communications, 2018: 329 – 334.
- [15] Draper-Gil G, Lashkari A H, Mamun M S I, et al. Characterization of encrypted and VPN traffic using time-related features[C]//2nd International Conference on Information Systems Security and Privacy, 2016: 407 – 414.
- [16] Lotfollahi M, Siavoshani M, Zade R, et al. Deep packet: A novel approach for encrypted traffic classification using deep learning[J]. Soft Computing, 2020, 24(3): 1999 – 2012.
- [17] 马丹, 万良, 程琪琴, 等. Attention-CNN 在恶意代码检测中的应用研究[J]. 计算机科学与探索, 2021, 15(4): 670 – 681.
- [18] Kingma D P, Ba J. ADAM: A method for stochastic optimization[EB]. arXiv:1412.6980, 2014.
- [19] 薛文龙, 于炯, 郭志琦, 等. 基于特征融合卷积神经网络的端到端加密流量分类[J]. 计算机工程与应用, 2021, 57(18): 114 – 121.