·安全技术· 文章编号：1000-3428(2018)11-0140-08 文献标志码：A 中图分类号：TP302

基于堆栈式自动编码器的加密流量识别方法

王 攀[1]，陈雪娇[2]

（1. 南京邮电大学 现代邮政学院，南京 210003；2. 南京信息职业技术学院 通信学院，南京 210023）

摘 要： 基于浅层机器学习的加密流量识别方法准确率偏低，在特征提取和选择方面耗时耗力。为此，提出一种基于堆栈式自动编码器（SAE）的加密流量识别方法。该方法利用 SAE 的无监督特性及在数据降维等方面的优势，结合多层感知机（MLP）的有监督分类学习，实现对加密应用流量的准确识别。考虑到样本数据集的类别不平衡性对分类精度的影响，采用 SMOTE 过抽样方法对不平衡数据集进行处理。实验结果表明，该方法各项性能指标均优于 MLP 加密流量识别方法，识别精确度和召回率以及 F1-Score 均可达到 99% 。

关键词： 加密流量识别；深度学习；堆栈式自动编码器；流量分类；多层感知机；卷积神经网络

中文引用格式： 王 攀，陈雪娇. 基于堆栈式自动编码器的加密流量识别方法[J]. 计算机工程，2018，44(11)：140-147，153.

英文引用格式： WANG Pan，CHEN Xuejiao. SAE-based encrypted traffic identification method[J]. Computer Engineering，2018，44(11)：140-147，153.

SAE-based Encrypted Traffic Identification Method

WANG Pan[1]，CHEN Xuejiao[2]

（1. School of Modern Posts，Nanjing University of Posts and Telecommunications，Nanjing 210003，China；
2. School of Communication，Nanjing Vocational College of Information Technology，Nanjing 210023，China）

【Abstract】 To solve the problem that encrypted traffic identification methods based on machine learning are low in accuracy and time-consuming and costly in feature extraction and selection，this paper proposes a Stacked Autoencoder（SAE）-based encrypted traffic identification method. The method utilizes the unsupervised characteristics of SAE and its advantages in dimensional reduction，combined with supervised classification learning of Multilayer Perceptron（MLP）to achieve accurate identification of encrypted application traffic. Considering the influence of the class imbalance of the sample dataset on the classifier performance，the unbalanced dataset is processed by the SMOTE oversampling method. Experimental results show that，the performance indicators of this method are higher than the MLP encrypted traffic identification method，and the precision，recall，and F1-Score can reach 99% .

【Key words】 encrypted traffic identification；deep learning；Stacked Autoencoder（SAE）；traffic classification；Multilayer Perceptron（MLP）；Convolutional Neural Network（CNN）

DOI： 10.19678/j.issn.1000-3428.0052059

0 概述

流量分类与识别是提升网络管理与安全监测水平，改善服务质量的基础，也是网络设计与规划等网络行为的前提。随着用户隐私保护和安全意识的增强，SSL、SSH、VPN 和 Tor 等技术得到了越来越广泛的应用，导致加密流量在网络传输中的比重越来越高。

因采用应用层加密，传统的端口匹配[1]、DPI 深度包检测[2-4]等技术无法准确识别这类加密应用流量，相关研究人员也一直在尝试通过各种机器学习方法，如 SVM、决策树、朴素贝叶斯等[5-7]，围绕流特征、净荷特征以及混合特征进行识别和分类。然而这些方法都存在 2 个问题，一个就是机器学习的特征提取和选择非常复杂，耗时耗力，且非常依赖特征专家的知识面和经验；另一个就是分类准确率不高。

相比其他机器学习方法，深度学习具有深层结构的特点，而其他浅层结构模型，如最大熵和 Softmax 回归、SVM 等方法的局限性主要是依赖于有标记的样本、对复杂函数难以表示和陷入局部最优化，或者泛化能力受到样本数据量的影响较大，如 BP 算法。深度学习采用训练多个单层非线性网络，组合底层特征构成数据的抽象表示，从而发现并且刻画问题内部复杂的结构特征，故而表达数据的本

基金项目： 江苏高校品牌专业建设工程项目（PPZY2015A092）。

作者简介： 王 攀（1979—），男，副研究员、博士，主研方向为深度学习、网络安全、信息网络；陈雪娇，讲师、硕士。

收稿日期： 2018-07-09 **修回日期：** 2018-08-10 **E-mail：** wangpan@njupt.edu.cn

质特征。

鉴于上述深度学习的优点,研究人员开始探讨深度学习方法在流量识别中的应用。文献[8]尝试用深度学习的方法来识别流量,采用堆栈式自动编码器(Stacked Autoencoder,SAE)来识别协议及应用,并对未知流量进行识别,达到了较好的准确率,但文中所采用的数据集为非公开数据集,无法进行验证对比。文献[9]提出采用 SAE 和卷积神经网络(Convolutional Neural Network,CNN)对加密流量进行识别,但对于数据的预处理和模型参数的选择等方面论述得不够清晰。文献[10]提出一种基于 CNN 的流量分类算法,分别采用公开数据集和实际数据集进行测试,并与传统分类方法相比,提高了流量分类的精度,减少了分类使用的时间,但并未涉及对加密流量的分类识别。文献[11]提出基于马尔科夫模型的半监督学习分类器进行流量分类。

本文提出一种基于 SAE 的加密流量识别方法,一方面利用 SAE 的无监督特性及在数据降维方面的优势,结合多层感知机(Multi-Layer Perceptron,MLP)的有监督分类学习,实现对加密应用流量的准确识别;另一方面考虑到样本数据集的类别不平衡性对分类精度的影响,本文采用 SMOTE 过抽样方法对不平衡数据集进行处理。

1　SAE

自动编码器是一种多层神经网络,属于非监督学习,不需要对训练样本进行标记,其输入层和输出层表示相同的含义——具有的节点数。自动编码器学习的是一个输入输出相同的“恒等函数”,其意义在于中间层,这一层是输入向量的特征表达。其原理表示如下:

假设输入一个 n 维信号 $x(x\in[0,1])$,经过输入层到达中间层,信号变为 y,可以用下式表示:

$$y=a(Wx+b) \tag{1}$$

其中,$a()$ 表示激活函数,如 Sigmoid 函数、ReLU 函数等,实现非线性变换,W 为权值,b 为偏置。信号 y 经过解码层解码,输出到输出层,输出层和输入层一样有 n 个神经元,假设输出信号为 z,计算公式如下:

$$z=a(W'y+b') \tag{2}$$

则 z 为 x 的预测值,W'、b' 同样分别表示权值与偏置,以区别于 W、b。通过调整网络参数,使得最终输出的 z 与原始输入的 x 尽可能相近。可通过损失/误差函数进行最小化运算得到,损失/误差函数可以是均方差(MSE)、交叉熵(Cross Entropy)等。

SAE 是深度学习领域常用的一个深度学习模型,由多个自动编码器堆叠而成,其目的是为了逐层提取输入数据的高阶特征,常被称为逐层贪婪训练学习过程,在此过程中逐层降低输入数据的维度,将一个复杂的输入数据转化成了一个系列简单的高阶特征,然后再把这些高阶特征输入一个分类器进行分类[12]。如图 1 所示,SAE 的训练过程主要分为以下 4 步:

1)给定初始输入 Y,采用无监督方式训练第一层自动编码器 V,输出为 Y',设定输入 Y 和输出 Y' 的损失函数,或者叫做重构误差,最小化损失函数以减少重构误差达到设定值。

2)把第一个自动编码器(AE1)隐含层的输出 V 作为第二个自动编码器(AE2)的输入,采用以上同样的方法训练自动编码器 Z。

3)重复第 2 步直到初始化完成所有自动编码器。

4)把最后一个 SAE 的隐含层输出(如图 1 中的 Z)作为分类器的输入,然后采用有监督的方法训练分类器的参数。

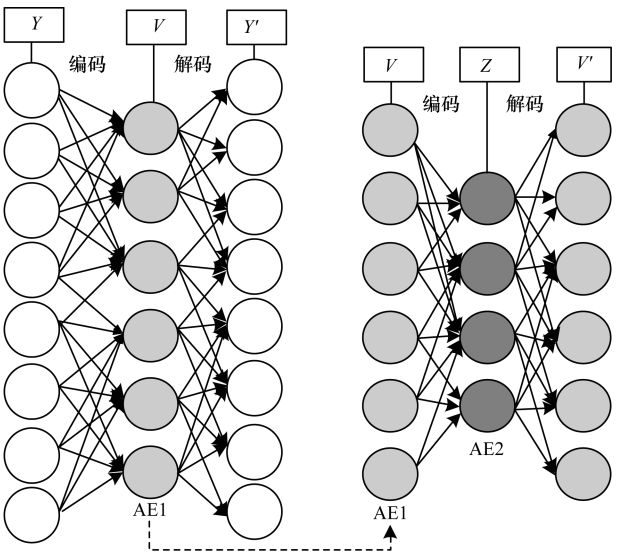


图 1　SAE 训练过程

2　基于 SAE 的加密流量识别方法

2.1　数据预处理

网络流量数据与图像既有相同之处,也有不同之处。相同之处在于网络数据包分组中的每一个字节都是由 8 bit 构成,取值范围为 0 ~ 255,非常类似于黑白图片中的一个灰度像素,可以借鉴很多图像识别的深度学习方法;不同之处在于数据包分组的长度,即每个数据包所包含的字节数长短不一,因而无法直接作为深度学习模型的输入,必须进行数据预处理。图 2 为样本数据的预处理过程。

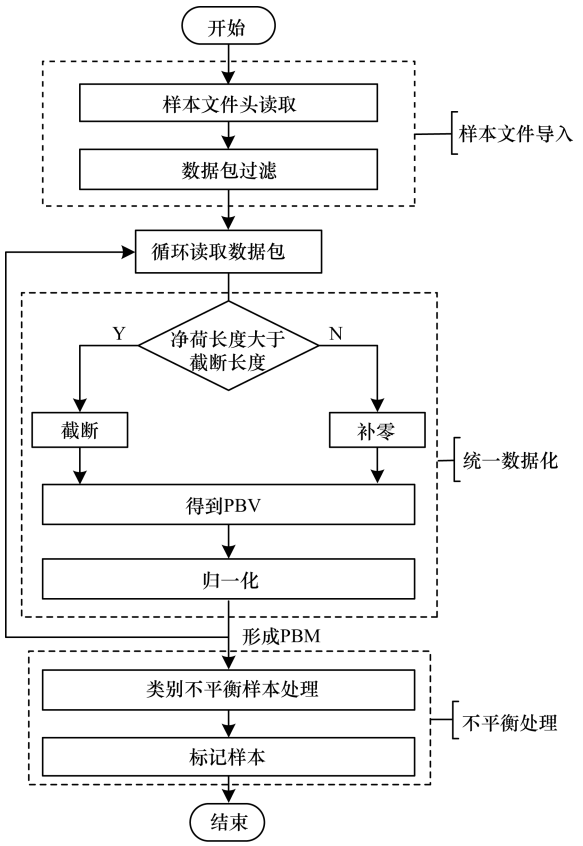


图 2 数据预处理流程

数据预处理流程如下:

步骤 1 样本数据包文件导入:流量样本数据集有各种形式,比如原始数据包、包特征文件等。为获取流量样本文件中所存储的分组信息,一般首先对样本文件头部进行读取,以获得该样本文件的概要信息,比如总共有多少分组或有哪些特征项。如果是原始数据包形式,还需要进行数据包过滤。原因在于样本数据包文件往往不够纯净,即其中虽包含大量的准确样本数据包,但也包含部分其他数据包,比如 APR、DHCP 等局域网数据,这些数据的滤除对于后期的训练和测试都是有意义的。

步骤 2 循环读取数据包:预处理过程逐个读取流量样本数据集中每一个分组,对该分组完成预处理之后再读取下一个分组,直至文件中最后一个分组为止。

步骤 3 分组截断(Truncation)和补零(zero-padding):形成分组字节矩阵(Packet Byte Matrix, PBM)作为深度学习模型的输入。其中,PBM 的每一行都是流量样本中的一个分组,也是网络的输入基本单元,称为分组字节向量(Packet Byte Vector, PBV)。考虑到深度学习的输入维度需统一,而每个分组的大小都不一样,因此需要考虑要么统一截断,要么长度不足的补零。定义 Truncation_len 为截断长度,需要考虑如下 3 个问题:

1)每个分组的大小最大值为 MTU,即 1 500 Byte,即以太网传输的最大 IP 报文(包含 IP 头部)是 1 500 Byte。所以,Truncation_len 值的理论范围为 0 Byte ~ 1 500 Byte。

2)Truncation_len 的选值。该值过大,会造成深度学习训练的输入参数过多,从而加大训练的复杂度;该值过小,则会牺牲识别的准确性,因为有可能被截断的净荷内容中是包含有流量识别的特征信息。

3)数据集中的分组大小分布。从数据集中的分组大小的分布可以看出,分组大小主要分布在头部(前 300 Byte)和尾部(后 1 200 Byte ~ 1 500 Byte)。因此,本文从保障准确性的角度出发,选择 Truncation_len 为最大值,即 1 500 Byte。

步骤 4 归一化(Normalization):为了提升训练的性能,将每个分组归一化至 0 ~ 1。每个分组大小为 0 ~ 255,因此每个分组大小均除以 256。

步骤 5 类别不平衡的样本数据集处理。详述见节 2.2。

步骤 6 样本标记:根据流量样本数据集对数据包文件的标记。对 PBM 进行标记,可以通过在矩阵中增加一列用于标记每一行,即 PBV,也可单独建立标记向量用于标记 PBV。

2.2 类别不平衡的样本数据集处理

无论是浅层机器学习,还是深度学习,分类识别研究基础都是基于一种假设:各种网络应用流都是均匀分布在网络中,即网络数据流的应用类别是平衡的。然而,现实网络中各种加密应用数据流分布很不均衡,比如通过加密协议承载的音、视频流远大于即时通信、纯网页加密流等。网络流类别不平衡是指流量样本数据集中存在的类别样本数量不均衡,通过训练,这些分类算法可能会忽略少数类别的流样本导致欠拟合,或重视少数类的差别造成过拟合[13]。

在进行深度学习训练之前,本文采用过抽样和欠抽样 2 种办法来处理不平衡数据集。对于比例过大的样本采用欠抽样方法来减少样本数量;对于比例过小的样本采用过抽样技术提高样本的可分性,拓展分类的决策边界。SMOTE 算法是一种随机过抽样方法,其主要思想是应用 K-最近邻方法,在少数类样本之间利用线性插值生成新的样本,从而增加少数类样本的数量,使数据集的类别数量相对平衡[14]。SMOTE 算法按照一定规则,利用插值方法合成新样本有良好的泛化作用,避免了随机复制原有样本的盲目和局限。没有引进少数类的冗余信息,扩展决策区域,避免过拟合问题,改善了数据集的可分性。

本文所采用的数据来源于“ISCX VPN-non VPN traffic dataset”[15],该数据集包括常规加密和 VPN 隧道传输的 2 类加密应用。本文选择常规加密流量中的 15 种应用作为训练和测试的样本数据,样本数据集均为 PCAP 文件格式。为了说明不平衡样本对性能的影响,同时给出了平衡样本,详细描述见表 1。从表 1 可以看出,不平衡样本中比例最高的 netflix 可以达到 27.701%,比例最少的 aim_chat 仅有 0.663%;对于样本量少的类别采用 SMOTE 过抽样方法,而对于样本量多的类别采用欠抽样方法,通过这 2 种方式预处理之后可以看出,样本基本都在 5%~8%。

表 1　样本数据集描述

应用名称	不平衡样本		平衡样本	
	数量	比例/%	数量	比例/%
AIM_chat	1 243	0.663	4 869	6.634
Email	4 417	2.356	4 417	6.018
Facebook	2 192	1.169	5 527	7.531
Gmail	2 329	1.242	5 000	6.813
Hangout	2 587	1.379	5 000	6.813
ICQ	1 986	1.059	4 243	5.781
Netflix	51 932	27.701	5 000	6.813
SCPdown	15 390	8.209	5 000	6.813
SFTPdown	4 729	2.523	4 729	6.443
Skype	4 607	2.457	4 607	6.277
Spotify	14 442	7.704	5 000	6.813
TorTwitter	14 654	7.816	5 000	6.813
Vimeo	18 755	10.004	5 000	6.813
Voipbuster	35 469	18.919	5 000	6.813
Youtuebe	12 738	6.794	5 000	6.813
总计	187 470	100.000	73 392	100.000

2.3　基于 SAE 的模型架构及加密流量识别算法

根据第 1 节所述的 SAE 模型的训练过程,可将基于 SAE 的加密流量识别模型的搭建和训练分为 2 大阶段:第一阶段是 SAE 的训练过程,这一深度学习训练过程是无监督的,称之为 SAE 训练过程;第二阶段是将训练好的 SAE 与分类输出相连,完成一个基于 MLP 的分类模型的训练,这一阶段是有监督的,称之为 SAE 分类器训练过程。基于上述的 2 个训练过程,本文所提出的 SAE 模型架构也分为 2 个部分:SAE 训练过程和 SAE 分类器训练过程。

如图 3 所示,第一阶段,SAE 训练过程中由 3 个自动编码器堆叠而成。

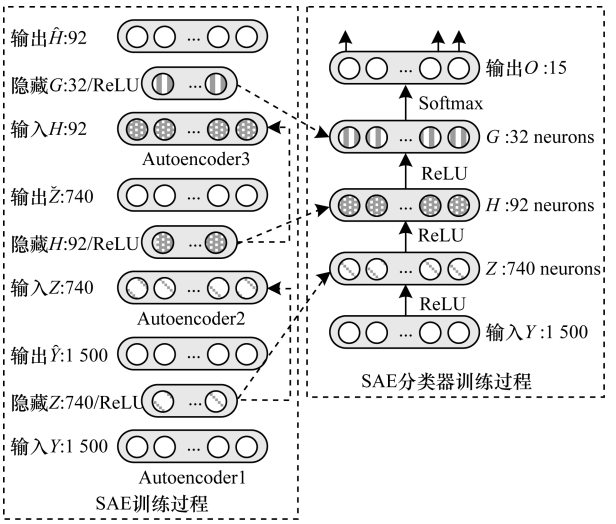


图 3　SAE 模型架构

训练过程分为如下几步:

步骤 1　第 1 个自动编码器(Autoencoder1),由一个输入层 Y,一个隐藏层 Z 和一个输出层 $\hat{Y}$ 组成。其中输入层来自 PBM 中的每一个分组,也就是矩阵中的每一个 PBV,根据上述的数据集预处理过程,训练数据输入维度为1 500;隐藏层采用 740 个神经元,对输入数据进行加权、偏置,即:

$$Z = \sum WY + b \tag{3}$$

计算后得到的结果采用 ReLu 作为激活函数,其公式表示如下:

$$ReLU(Z) = \max[0, Z] \tag{4}$$

输出层与输入层相同,由 1 500 个神经元构成。定义损失函数为 MSE,为尽可能减少重构误差,实质上就是最小化输入与输出的误差,假设 n 为输出神经元个数,则定义如下:

$$\min(M_{\mathrm{MSE}}) = \min\left(\frac{1}{n}\sum_{i=1}^{n}(Y_i - \hat{Y}_i)^2\right) \tag{5}$$

在反向传播的过程中,采用随机梯度下降(Stochastic gradient descent,SGD)算法更新权值 W、偏置 b 以最小化损失函数,计算公式如下:

$$W \leftarrow W - \rho\frac{\partial L}{\partial W} \tag{6}$$

$$b \leftarrow b - \rho\frac{\partial L}{\partial b} \tag{7}$$

其中,ρ 为学习速率。

步骤 2　第 2 个自动编码器(Autoencoder2),也是由一个输入层,一个隐藏层和一个输出层组成。根据 SAE 逐层贪婪训练的特点,Autoencoder2 的输入采用 Autoencoder1 训练好的隐藏层,也即图 3 中的 Z,所以输入的神经元个数为 740;隐藏层为 92 个神经元,ReLu 为激活函数,输出 $\hat{Z}$ 神经元个数与输入 Z 相同为 740。

步骤 3 第 3 个自动编码器(Autoencoder3),同步骤 2,采用 Autoencoder2 的隐藏层 H 作为输入,神经元个数为 92;隐藏层 G 为 32,激活函数为 ReLU,输出 $\hat{H}$ 神经元个数与输入 H 相同为 92。

第 1 阶段整体处理过程总结见算法 1,即 SAE 模型训练算法。

算法 1 SAE 模型训练算法

输入 $Y=\{y_1,y_2,\cdots,y_k\}$,$k=1\ 500$,即 PBV

输出 $\hat{Y}=\{\hat{y}_1,\hat{y}_2,\cdots,\hat{y}_k\}$

1. 设定相关初始化参数。激活函数 α = ReLU; e 为训练周期,即 epoches;τ 为 mini_batch 的大小;
2. for t from 1 to e do
3. for each τ do
4. 针对每一个流量样本 $y\in Y$,执行如下步骤:
5. 前向传播:计算式(3);
6. 激活函数:计算式(4);
7. 计算损失函数并最小化:计算式(5);
8. 反向传播更新权值:计算式(6)和式(7);
9. end for
10. end for

第 2 个阶段是 SAE 分类器训练过程,如图 3 所示,由 1 个输入层、3 个隐藏层和 1 个输出层构成,训练过程分为如下 6 步:

步骤 1 输入数据采用第 2.1 节所描述的数据预处理后所得到的 PBV,训练数据输入维度为 1 500,即输入为 1 500 个神经元。

步骤 2 3 个隐藏层分别使用 SAE 模型中训练好的 Autoencoder1、Autoencoder2 和 Autoencoder3,神经元个数分别为 740、92 和 32,对输入数据进行加权、偏置,见式(3)。

步骤 3 计算后得到的结果均采用 ReLu 作为激活函数,见式(4)。

步骤 4 输出层为 15 个神经元,分别代表 15 个加密应用类别,采用 Softmax 激活函数进行分类,假设 z 为 Softmax 输入,$\hat{y}$ 表示输出,则公式表示如下:

$$\hat{y}_j=\frac{\exp(z_j)}{\sum_1^n\exp(z_i)} \tag{8}$$

步骤 5 在模型训练过程中,采用交叉熵作为损失函数,假设 L 表示损失函数,则定义如下:

$$L=-\frac{1}{n}\sum_1^n\hat{y}_i\ln f(x,W,b) \tag{9}$$

步骤 6 在反向传播的过程中,同样采用 SGD 算法更新权值 W、偏置 b 以最小化损失函数,见式(6)和式(7)。

第 2 阶段整体处理过程总结见算法 2,即 SAE 分类模型训练算法。

算法 2 SAE 分类模型训练算法

输入 $Y=\{y_1,y_2,\cdots,y_k\}$,$k=1\ 500$,即 PBV

输出 $\hat{O}=\{\hat{o}_1,\hat{o}_2,\cdots,\hat{o}_j\}$,$j=15$,即加密应用类别个数

1. 设定相关初始化参数。隐藏层激活函数 α = ReLU;分类激活函数 s = Softmax;e 为训练周期,即 epoches;τ 为 mini_batch 的大小;
2. for t from 1 to e do
3. for each τ do
4. 针对每一个流量样本 $y\in Y$,执行如下步骤:
5. 前向传播:计算式(3);
6. 激活函数:计算式(4);当到达输出层时,计算式(8);
7. 计算损失函数并最小化:计算式(9);
8. 反向传播更新权值:计算式(6)和式(7);
9. end for
10. end for

3 实验与结果分析

3.1 实验设置

为了测试并对比 SAE 的流量识别性能,本文选择最基本的深度学习模型——MLP 来与之对比,实验环境参数见表 2。本文设计了一个简单的 MLP 模型用于加密流量识别,模型采用 1 个输入层,1 500 个神经元;2 个隐藏层,分别为 128 和 32 个神经元,激活函数为 ReLU;1 个输出层,15 个神经元,激活函数为 Softmax 用于分类。

表 2 实验环境参数

类别	参数
处理器	Nvidia GPU(GeForce GTX 1080)
操作系统	Windows 10,64 bit
深度学习平台	Keras[16]
深度学习后端	TensorFlow-gpu 1.4.0[17]
CUDA 版本	8.0
CuDNN 版本	6.0

在训练过程中,将数据集随机的分成 2 部分:一部分是训练集,占比为 60%;剩下的 40% 为测试集。深度学习的模型训练过程中所采用的优化器、损失函数、epoches 以及 mini_batch 等参数信息见表 3。

表 3 模型训练参数设置

训练参数	优化器	损失函数	Epoches	Mini_batch
MLP 分类	Adam	分类交叉熵	200	256
SAE 分类	Adadelta	MSE	200	128
SAE 分类	adam	分类交叉熵	200	256

图 4 为 SAE 模型的训练过程中准确率和损失率的变化趋势。可以看出,在 200 个训练周期中,SAE 模型的准确率稳定攀升,直至 0.99,最后趋于稳定;而损失率持续下降并趋于稳定。

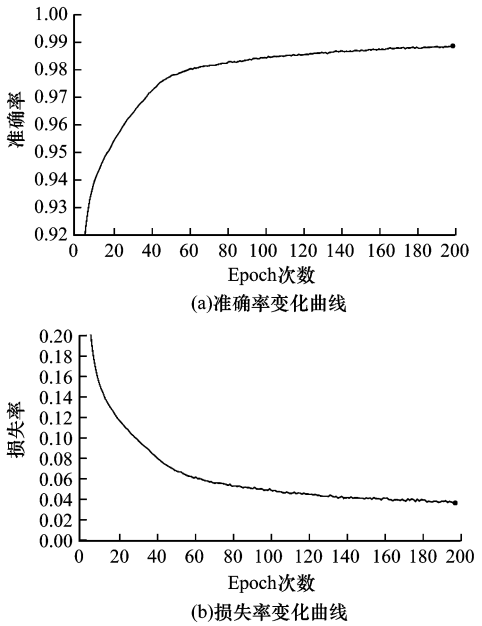


图 4　训练过程中的准确率和损失率变化曲线

3.2　性能参数

为了评估模型的性能,本文采用以下 2 类指标:

1)精确率和召回率

误报(False Positive,FP)是指非类别 C(C 指代一个特定的类别)的流量被分类成为类别 C;真阴性(True Negative,TN)是指非类别C的流被分成非类别 C;漏报(False Negative,FN)是指属于类别 C 的流量被分类为非类别 C;真阳性(True Positive,TP)是指属于类别 C 的流量而被分类成类别 C。

精确率(Precision)和召回率(Recall)计算如下:

$$Precision = \frac{TP}{TP + FN} \tag{10}$$

$$Recall = \frac{TP}{TP + FP} \tag{11}$$

2)F1-Score

F1-Score 是精确率和召回率加权调和平均,用于综合反映整体的指标[18],最常见的 F1-Score 计算公式为:

$$F1 = \frac{2 \cdot Precision \cdot Recall}{Precision + Recall} \tag{12}$$

3.3　基于平衡与不平衡数据集的对比

为了验证不平衡数据集与平衡数据集对分类性能的影响,本文分别用节 2.2 中所描述 2 种数据集,采用基于 MLP 的深度学习方法进行模型训练和测试,得出的分类混淆矩阵如图 5 所示。在该矩阵中,对角线上的元素代表分类正确的样本数量,其他元素均为误判。可以非常明显地看出,图 5(a)中有不少分类错误的样本,而图 5(b)中分类性能较好。可以看出,不平衡数据集对于分类器的性能有直接的影响,因此本文中后续的实验均基于第 2.2 节所描述的平衡数据集,以期最大程度上减少对分类性能的影响。

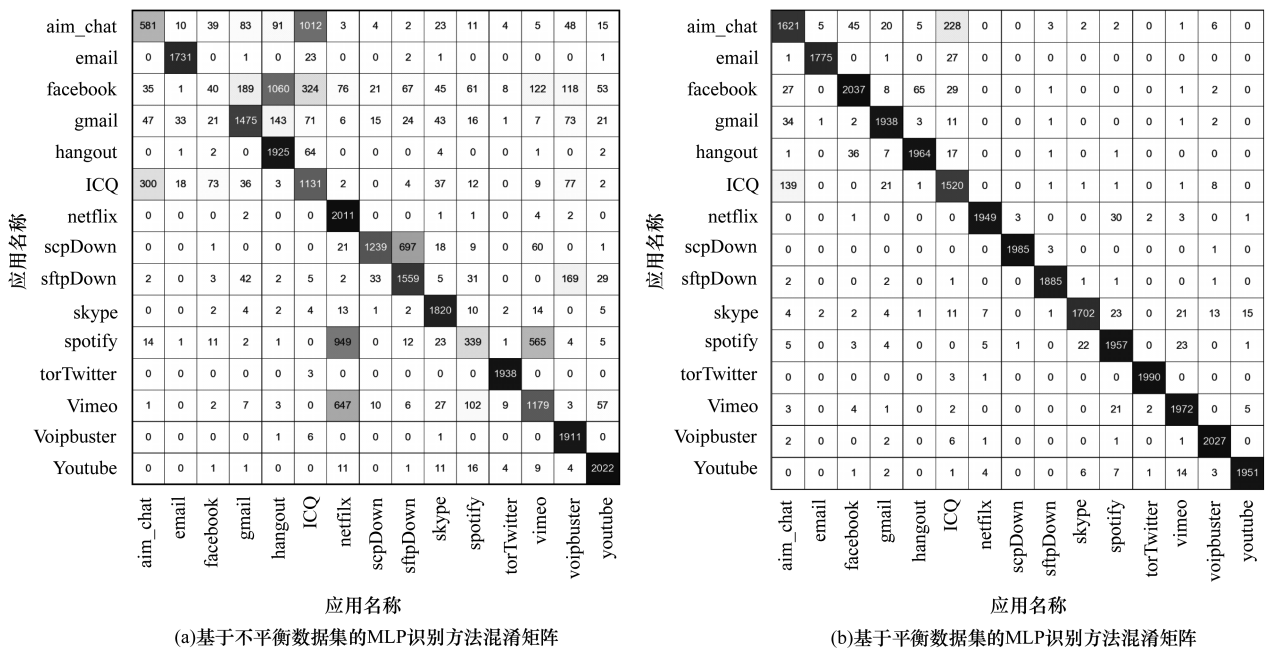
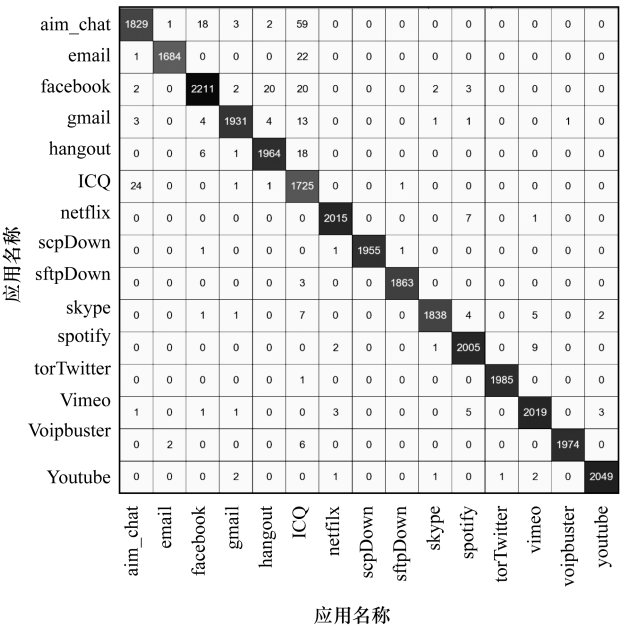


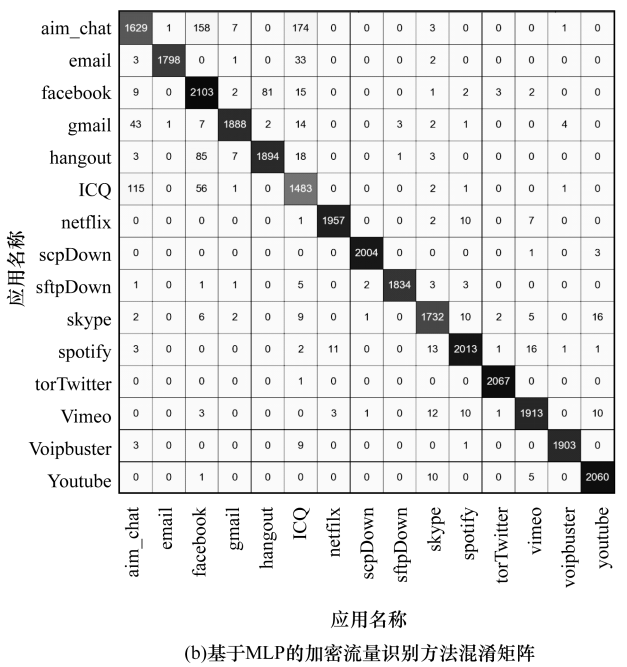
图 5　基于平衡、不平衡数据集的 MLP 识别方法混淆矩阵

3.4 分类实验结果

图 6 展示了基于平衡数据集训练出来的 MLP 和 SAE 2 种加密流量识别方法的分类混淆矩阵。从对角线上的元素数量,也即分类正确的样本数量来看,基于 SAE 的分类方法要高于 MLP 方法。比如,对于 aim_chat 与 ICQ 这 2 种加密应用而言,因均为聊天类的应用,所以流量特征方面较为相近,容易造成误判。在 MLP 识别方法中,aim_chat 被误判为 ICQ 的样本数量高达 174,已接近测试集的 10%;而在 SAE 识别方法中,aim_chat 被误判为 ICQ 的样本数量仅为 59,大致为测试集的 3%。



(a)基于SAE的加密流量识别方法混淆矩阵



(b)基于MLP的加密流量识别方法混淆矩阵

图 6 基于 SAE、MLP 的加密流量识别方法混淆矩阵

从混淆矩阵中还不能直观的反映出 2 种方法的分类性能,图 7 和图 8 分别展示了 MLP 和 SAE 2 种加密流量识别方法在精确度、召回率和 F1-Score 3 个指标上的对比。虽然基于 MLP 的加密流量识别方法显示出比较好的性能,但可以明显看出,在 aim_chat、Facebook、hangout 和 ICQ 这 4 种流量特征相近的即时通信类加密应用的分类性能上,有比较大的差距。而图 8 所展示出的基于 SAE 的加密流量识别方法则展示出了很好的分类性能。从表 4 中更能清晰地看到每一种加密应用的 3 项指标,尤其是总体指标,可以看出,基于 SAE 的加密流量识别方法在分类精度、召回率和 F1-Score 3 项指标均达到 99%,高于基于 MLP 的加密流量识别方法的指标(96%),显示出了优秀的分类性能。

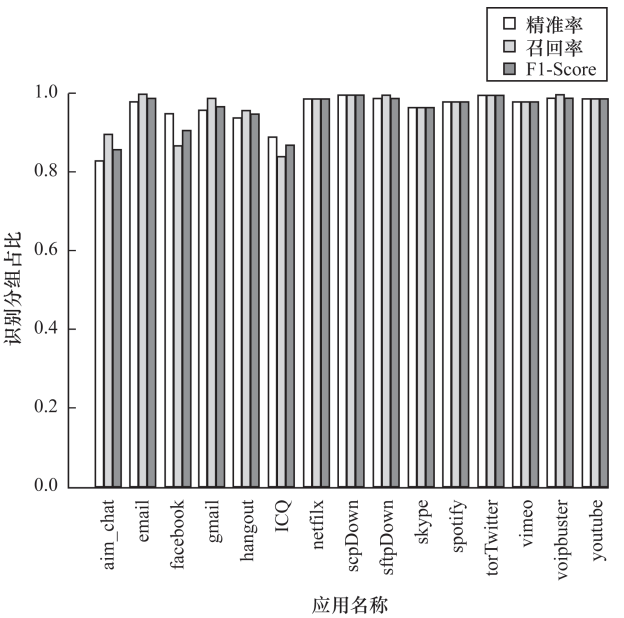


图 7 基于 MLP 方法的精度、召回率和 F1-Score

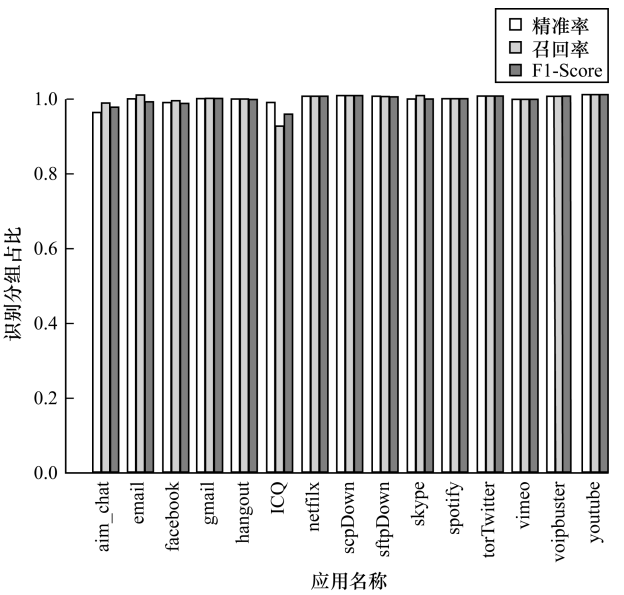


图 8 基于 SAE 方法的精度、召回率和 F1-Score

表 4　SAE、MLP 的精确度、召回率和 F1-Score

应用名称	SAE			MLP		
	精确度	召回率	F1-Score	精确度	召回率	F1-Score
AIM_chat	0.98	0.96	0.97	0.90	0.83	0.86
Email	1.00	0.99	0.99	1.00	0.98	0.99
Facebook	0.99	0.98	0.98	0.87	0.95	0.91
Gmail	0.99	0.99	0.99	0.99	0.96	0.97
Hangout	0.99	0.99	0.99	0.96	0.94	0.96
ICQ	0.92	0.98	0.95	0.84	0.89	0.87
Netflix	1.00	1.00	1.00	0.99	0.99	0.99
SCPdown	1.00	1.00	1.00	1.00	1.00	1.00
SFTPdown	1.00	1.00	1.00	1.00	0.99	0.99
Skype	1.00	0.99	0.99	0.97	0.97	0.97
Spotify	0.99	0.99	0.99	0.98	0.98	0.98
TorTwitter	1.00	1.00	1.00	1.00	1.00	1.00
Vimeo	0.99	0.99	0.99	0.98	0.98	0.98
Voipbuster	1.00	1.00	1.00	1.00	0.99	0.99
Youtube	1.00	1.00	1.00	0.99	0.99	0.99
平均	0.99	0.99	0.99	0.96	0.96	0.96

4　结束语

本文提出一种基于 SAE 的深度学习方法对加密应用流量进行分类识别,利用 SAE 的无监督特性及在数据降维方面的优势,并结合 MLP 的有监督分类学习,实现对加密流量的准确识别。同时考虑到样本数据集的类别不平衡性对分类精度的影响,本文采用 SMOTE 过抽样方法对不平衡数据集进行处理。实验结果表明,该方法在样本数据集类别平衡的情况下识别精确度和召回率可以达到 99% 。在未来工作中,将进一步减少训练参数,并研究采用非监督的方式对加密应用实现准确识别。

参考文献

[1] TOUCH J, KOJO M, LEAR E, et al. Service name and transport protocol port number registry [EB/OL]. [2018-06-06]. http://www.iana.org/assignments/port-numbers.

[2] KHALIFE J, HAJJAR A, DIAZ-VERDEJO J. A multilevel taxonomy and requirements for an optimal traffic-classification model[J]. International Journal of Network Management, 2014, 24(2): 101-120.

[3] PARK B C, WON Y J, KIM M S, et al. Towards automated application signature generation for traffic identification[C]//Proceedings of Network Operations and Management Symposium. Washnington D. C., USA: IEEE Press, 2008: 160-167.

[4] SHERRY J, LAN C, POPA R A, et al. Blindbox: deep packet inspection over encrypted traffic [C]// Proceedings of ACM Conference on Special Interest Group on Data Communication. New York, USA: ACM Press, 2015: 213-226.

[5] 陈　伟,胡　磊,杨　龙. 基于载荷特征的加密流量快速识别方法[J]. 计算机工程,2012,38(12):22-25.

[6] MENG P, ZHOU G, MENG J. Fast identification of encrypted traffic via large-scale sparse screening[C]// Proceedings of International Conference on Advanced Cloud & Big Data. Washington D. C., USA: IEEE Press, 2017: 273-278.

[7] OKADA Y, ATA S, NAKAMURA N, et al. Comparisons of machine learning algorithms for application identification of encrypted traffic [C]//Proceedings of International Conference on Machine Learning and Applications and Workshops. Washington D. C., USA: IEEE Press, 2011: 358-361.

[8] WANG Z. The applications of deep learning on traffic identification[C]//Proceedings of Black Hat 2015. Singapore: [s. n.], 2015.

[9] LOTFOLLAHI M, ZADE R S H, SIAVOSHANI M J, et al. Deep packet: a novel approach for encrypted traffic classification using deep learning[J/OL]. [2018-06-06]. http://cn.arxiv.org/pdf/1709.02656v3.

[10] 王　勇,周慧怡,俸　皓,等. 基于深度卷积神经网络的网络流量分类方法[J]. 通信学报,2018,39(1):14-23.

[11] 赵　英,韩春昊. 马尔科夫模型在网络流量分类中的应用与研究[J]. 计算机工程,2018,44(5):291-295.

[12] CAUDILL M. Neural networks primer, part I[J]. AI Expert, 1987, 2(12): 46-52.

(下转第 153 页)

4　结束语

SM4 作为我国自行设计的分组密码标准,采用 32 轮非线性迭代结构,具有很高的安全性,但仍然面临 DPA 攻击的巨大风险。现有的防护 DPA 攻击的掩码方案虽然很多,但多数局限性较强,且防护能力较弱。本文通过对秘密共享抵抗 DPA 攻击的原理分析及文献[8]提出的 AES 算法 S 盒共享实现方案的研究,构造了一个适用于 SM4 算法实现的新型共享 S 盒,新的 S 盒通过利用秘密共享函数代替仿射变换,在乘法器分组中采用虚拟值法,并在反相器中引入分解法,使得本文实现方案具有较少的运算次数和较低的空间占比。安全性分析和实验结果表明,该方案对高阶 DPA 攻击乃至 glitch 攻击具有较强的抵抗能力。

参考文献

[1] 吕述望,苏波展,王　鹏,等. SM4 分组密码算法综述[J]. 信息安全研究,2016(11):995-1007.

[2] LIU F,JI W,HU L,et al. Analysis of the SM4 block cipher[C]//Proceedings of ACISP'07. Townsville, Australia:[s. n.],2007:158-170.

[3] BAI X,GUO L,LI T. Differential power analysis attack on SM4 block cipher[C]//Proceedings of ICCSC'08. Washington D. C.,USA:IEEE Press,2008:613-617.

[4] BAI X, XU Y, GUO L. Securing SM4 cipher against differential power analysis and its VLSI implementation[C]//Proceedings of the 11th IEEE International Conference on Communications Systems. Washington D. C., USA:IEEE Press,2008:167-172.

[5] LIANG H,WU L,ZHANG X,et al. Design of a masked S-box for SM4 based on composite field[C]//Proceedings of the 20th International Conference on Computational Intelligence and Security. Washington D. C., USA: IEEE Press,2014:387-391.

[6] NIKOVA S,RECHBERGER C,RIJMEN V. Threshold implementations against side-channel attacks and glitches[C]//Proceedings of International Conference on Information and Communications Security. Berlin, Germany:Springer,2006:529-545.

[7] MORADI A,POSCHMANN A,LING S,et al. Pushing the limits: a very compact and a threshold implementation of AES[C]//Proceedings of Advances in Cryptology-EUROCRYPT'11. Berlin, Germany: Springer,2011:69-88.

[8] BILDIN B,GIERLICHS B,NIKOVA S,et al. A more efficient AES threshold implementation [C]// Proceedings of International Conference on Cryptology. Berlin,Germany:Springer,2014:267-284.

[9] KOCHER P C, JAFFE J, JUN B. Differential power analysis[C]//Proceedings of International Cryptology Conference on Advances in Cryptology. Berlin, Germany:Springer,1999:388-397.

[10] MANGARD S, OSWALD E, POPP T. Power analysis attacks:revealing the secrets of smart cards[M]. Berlin, Germany:Springer,2010.

[11] SLINKO A. Secret sharing [M]. Berlin, Germany: Springer,2015.

[12] NIKOVA S, RIJMEN V, SCHLAFFER M. Secure hardware implementation of nonlinear functions in the presence of glitches[J]. Journal of Cryptology, 2011, 24(2):292-321.

[13] 冷建伟,李　鹏. 基于自适应特征分布更新的压缩跟踪算法[J]. 计算机工程,2018,44(2):264-270.

[14] WANG Y,YUAN Z,LI Z,et al. Secret sharing based countermeasure for AES S-box[C]//Proceedings of IEEE International Symposium on Integrated Circuits. Washington D. C.,USA:IEEE Press,2011:504-507.

[15] BILGIN B,NIKOVA S,NIKOVA V,et al. Threshold implementations of small S-boxes[J]. Cryptography and Communications,2015,7(1):3-33.

[16] 钟卫东,孟庆全,张帅伟,等. 基于秘密共享的 AES 的 S 盒实现与优化[J]. 工程科学与技术,2017(1):191-196.

[17] 袁　征. 功耗攻击防御技术在分组密码中的应用研究[D]. 长沙:湖南大学,2012.

[18] 牛砚波,蒋安平. 一种低功耗抗差分功耗分析攻击的 SM4 算法实现[J]. 微电子学与计算机,2014,31(9):28-32.

编辑　索书志

(上接第 147 页)

[13] 陈雪娇,王　攀,刘世栋. 网络应用流类别不平衡环境下的 SSL 加密应用流识别关键技术[J]. 电信科学,2015,31(12):83-89.

[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(1): 321-357.

[15] DRAPER-GIL G, LASHKARI A, MAMUN M, 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. Setúbal, Portugal: Science and Technology Publications,2016,407-414.

[16] GitHub,Inc. Keras:deep learning for humans [EB/OL]. [2018-06-06]. https://github.com/fchollet/keras.

[17] ABADI M,AGARWAL A,BARHAM P,et al. TensorFlow: large-scale machine learning on heterogeneous systems [EB/OL]. [2018-06-06]. http://cn.arxiv.org/pdf/1603.04467v1.

[18] KIM H, CLAFFY K, FOMENKOV M, et al. Internet traffic classification demystified: myths, caveats, and the best practices [C]//Proceedings of the 2008 ACM CoNEXT Conference. New York, USA: ACM Press, 2008:1-12.

编辑　刘盛龄