计算机科学与探索
Journal of Frontiers of Computer Science and Technology
1673-9418/2024/18(03)-0805-13
doi: 10.3778/j.issn.1673-9418.2304045

融合SENet和Transformer的应用层协议识别方法

陈　乾,洪　征+,司健鹏
中国人民解放军陆军工程大学 指挥控制工程学院,南京 210000
+ 通信作者 E-mail: hz5215@163.com

摘　要:协议识别技术在网络通信和信息安全领域具有至关重要的地位和作用。针对现有基于时空特征的协议识别方法提取协议特征不充分、不全面的问题,提出了一种基于SENet和Transformer的应用层协议识别方法。该方法关注协议数据的时空特征,由加入SENet注意力的残差网络构成的空间特征提取模块和Transformer网络编码器构成的时间提取模块组成。空间特征提取阶段,在残差网络结构中加入SE块获取多个卷积通道间的联系,自适应地为通道分配权重,提取不同通道中更加活跃的协议空间特征;时间特征提取阶段,由基于多头注意力机制的Transformer编码器通过堆叠的方式构建时间特征提取模块,利用输入数据的位置信息全面地获取协议数据的时间特征。通过对更加充足的空间特征和更加全面的时间特征的提取和学习,可以获得更有效的协议识别信息,进而提高协议识别性能。在ISCX2012和CSE_CIC_IDS2018混合数据集上的实验结果表明,所提模型的总体识别准确率达到99.20%, *F*1值达到98.99%,高于对比模型。

关键词:SENet;残差网络;自注意力;Transformer;协议识别;网络安全
文献标志码:A　　**中图分类号:**TP398.08

Application Layer Protocol Recognition Incorporating SENet and Transformer

CHEN Qian, HONG Zheng+, SI Jianpeng
Command and Control Engineering College, Army Engineering University of PLA, Nanjing 210000, China

Abstract: Protocol recognition technology assumes a crucial position and exerts significant influence in the domains of network communication and information security. Existing protocol recognition methods based on spatio-temporal features cannot adequately and comprehensively extract protocol features. An application layer protocol recognition method incorporating SENet channel attention and Transformer is proposed. The model focuses on spatio-temporal feature extraction of protocol data, and the model consists of a spatial feature extraction module and a time extraction module. SE blocks are added to the residual network to capture the associations between multiple channels and adaptively assign weights, so as to extract the key space features in different channels. The temporal feature extraction module is constructed by stacking the transformer encoders based on multi-head attention mechanism. This module is used to comprehensively capture temporal features of the protocol data by directly leveraging the positional information of the input data. After extracting and learning more detailed spatial features and more comprehensive temporal features, better protocol feature representation is obtained to improve protocol recognition performance. Experiments are conducted on the ISCX2012 and CSE_CIC_IDS2018 hybrid datasets, and the results show that the overall recognition accuracy of the proposed model reaches 99.20%, and the *F*1 score reaches 98.99%, which are higher than those of the comparison models.

Key words: SENet; residual network; self-attention; Transformer; protocol recognition; network security

基金项目:国家重点研发计划(2019YFB2101704)。
This work was supported by the National Key Research and Development Program of China (2019YFB2101704).
收稿日期:2023-04-13　　**修回日期:**2023-07-11

网络协议是网络中不同通信实体进行数据交换的基础。协议识别技术通过提取网络流量数据中的关键特征并进行分析,判定流量所属的应用层协议。协议识别有助于对流量的结构进行分析,为网络的正常运行提供保障[1],在网络流量监测、服务质量管理和安全检测等领域得到了广泛的应用。

依据分析粒度,协议识别技术可以分为基于数据包的识别和基于数据流的识别。基于数据包的识别以单个数据包的信息为基础,常常应用于在线协议识别领域。由于单个数据包中包含的信息较少,识别的准确率往往较低。基于数据流的识别将多个数据包依据通信对象和通信时间汇聚成网络流,能够更为精确地对通信情况进行判断。本文研究采用的是基于数据流的方法。

基于数据流的协议识别方法主要可以划分为基于固定规则的识别方法、基于主机行为的识别方法、基于传统机器学习的识别方法和基于深度学习的识别方法。基于固定规则的识别方法根据网络通信的五元组对协议进行识别[2],随着动态端口技术的广泛应用,此类方法识别准确率明显降低。基于主机行为的识别方法利用网络流量数据的统计特征进行识别,但复杂的网络环境会影响该方法所依赖的统计数据,导致误判和漏判。基于传统机器学习的识别方法依赖于特征工程,特征的选择对于协议识别的准确性有着很大影响[3]。深度神经网络可以自动化对协议数据的深层特征进行提取,避免人工进行特征选择的主观性,目前在协议识别领域应用较为广泛。

基于深度学习的协议识别方法所利用的协议数据特征主要包括空间特征和时间特征。同一种协议的消息序列中包含大量相同的固定字段,使相同协议产生的内容高度相似,此类文本信息可以作为协议的空间特征。协议数据还具有时间特征,通过协议状态反映协议的时序结构。

现有的协议识别方法可以分为基于单一特征的协议识别方法和基于混合特征的协议识别方法。基于单一特征的识别方法关注协议数据某个维度(空间或时间)的特征,单个维度的特性能够在一定程度上对协议进行表征。基于混合特征的协议识别方法采用多模型融合的方式实现,兼顾协议的空间特征和空间特征,较为全面地对协议进行建模,提高协议识别的准确率。

针对神经网络层次加深导致的梯度消失和梯度爆炸等问题,He 等人[4]提出了深度残差网络(deep residential network,ResNet)的概念,但是ResNet进行特征提取时,无法依据卷积通道之间的关系分配权重,会导致模型对协议的关键特征学习不足。

注意力机制(attention mechanism)能够选择特定区域着重学习或分配不同的权重,以此筛选并学习重要信息[5]。基于此,Hu 等人[6]提出的SENet通过SE块(squeeze-and-excitation block,SE block)对通道之间的依赖关系显式建模实现通道级的注意力机制,自适应地调整卷积通道的权重,为重要的通道分配大的权重,使得重要通道中的信息能够得到更充分的学习。

本文采用ResNet和SE块相组合的方式提取协议数据的空间特征。一方面,ResNet能够对关键特征进行提取,残差结构的引入可以使这些关键特征得到重复利用。另一方面,通过加入SE块能够充分考虑到通道级别的特征选取,细致获取协议特征。

在时间特征的提取上,以循环神经网络(recurrent neural network,RNN)为代表的序列模型基于输入序列中相邻单元的位置信息进行时间特征的学习,每个位置的分析都依赖于其之前单元的计算结果[7]。然而,这种运算规则无法直接计算不相邻单元之间的关联,而且随着单元间距的增大,前置单元信息对后置单元的影响越来越微弱。在分析长序列时,单元位置对特征学习有很大影响,不考虑不相邻单元之间的联系会导致时间特征学习不充分。协议数据流多为长序列,在处理协议数据流的时间特征时,需要从全局角度对数据进行特征学习,关注不相邻单元之间的关联。

Vaswani 等人[8]提出的Transformer结构在自然语言处理和图像领域取得了很好的效果。该模型通过多头注意力机制(multi-head attention)获取数据单元之间的相关系数来计算数据单元之间的权重。Transformer的输入是由多个单元组成的序列,其中每个单元包含自身位置信息。通过多头注意力学习位置信息,可以从全局的角度对输入数据进行建模,反映数据单元之间的联系。

基于上述分析,本文提出了基于SENet和Transformer的应用层协议识别方法。SENet用于提升空间特征的提取能力,在ResNet中插入SE块使得空间提取模块能够通过通道注意力学习提取更为细致的协议特征。基于Transformer的时间特征提取模块能

够从全局的角度对协议的时间特征进行提取。两者的结合可以有效提升协议识别效果。

1　相关工作

基于单一特征的协议识别方法关注协议数据的空间特征或时间特征。Feng等人[9]提出了PrtCNN模型，以协议数据的空间特征作为基础，将卷积神经网络用于协议识别领域。通过卷积神经网络对流量位图的空间特征进行学习，用于协议类型的判别。Wei等人[10]以协议数据的时间特征作为检测基础，基于长短期记忆（long short-term memory，LSTM）模型提出了ABL-TC（attention-based LSTM for traffic classification）协议识别模型。将序列化的协议数据作为模型输入，提取协议数据的时间特征用于协议识别。以上方法存在特征提取不充分，协议识别准确率偏低等问题。

基于混合特征的协议识别方法既考虑到协议数据的空间特征，也兼顾协议数据的时间特征。吴吉胜等人[11]将一维预激活残差网络（pre-activation ResNet, PreResNet）和双向门控循环神经网络（bi-directional gated recurrent units，BiGRU）进行组合，首先由一维PreResNet对协议数据的空间特征进行提取，接着利用BiGRU提取协议时间特征，最后利用注意力机制对关键特征进行学习以实现协议识别。该方法忽略了残差网络中卷积通道的关联对于特征学习的影响；在时间特征学习过程中虽采用注意力机制抑制了状态间隔对时序特征提取的影响，但仍然存在循环神经网络对全局性特征的学习存在局限的问题。Sarhangian等人[12]将卷积神经网络（convolutional neural network，CNN）与LSTM、门控循环单元（gated recurrent unit，GRU）分别进行融合，利用CNN对协议数据的空间特征进行提取，再由序列模型提取时间特征。空间特征提取采用传统卷积网络，缺乏提取通道级关键特征的能力。基于LSTM和GRU的时间特征提取结构在特征提取过程中主要关注相邻单元信息，不具备对全局性特征学习的能力，易造成误判。彭瑶[13]提出的STF-SA（spatio-temporal features and self-attention）模型基于时空特征对加密协议进行识别，首先通过卷积结构提取流量数据的空间特征，接着将空间特征向量中每个元素进行位置编码，通过多层叠加的时间特征提取模块进行时间特征的学习。该方法的空间提取模块通过多层卷积的叠加实现，难以对协议关键特征实现充分学习。另外，时间提取模块的输入序列相对较长，计算开销较大。

总体上看，现有的基于混合特征协议识别方法主要存在两方面的不足：其一，现有方法的空间特征提取往往依赖于卷积操作。卷积结构无差别地将多个卷积通道中的空间信息进行融合，忽略了卷积通道之间的关系。其二，时间特征的提取依赖于数据单元在输入序列中的位置。状态的计算依赖于相邻单元，模型在计算当前状态时对上一个状态有着较强依赖，当状态序列较长时，难以直接建立较早状态与较晚状态之间的关联，导致时间特征学习不全面。

现有基于混合特征的方法存在协议特征提取不充分、不全面的问题，为了更加有效地提取协议特征，本文提出了基于SENet通道注意力和Transformer的应用层协议识别方法。SENet根据卷积通道中的信息重要性为卷积通道分配权重，确保关键通道中的信息得到细致学习。基于Transformer的时间特征提取模块能够直接利用输入数据的全局信息表示协议单元之间的全局关系，全面获取协议的时间特征。最终所得混合特征既包含了关键的空间特征，也包含了全局性时间特征，有助于提升协议识别效果。

2　基于SENet和Transformer的应用层协议识别方法

2.1　工作流程

协议识别方法的工作流程主要包括数据准备阶段、模型训练阶段和模型测试阶段，如图1所示。数据准备阶段分为数据采集、数据预处理和数据集划分三个步骤。数据采集步骤使用Wireshark抓包工具进行原始流量的收集，获取需要分析的数据包。在

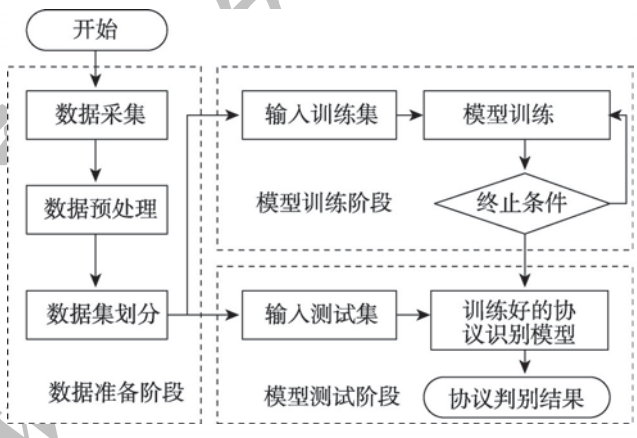

图1　协议识别的工作流程

Fig.1　Protocol recognition workflow

数据预处理步骤,将对数据包进行清洗,滤除干扰应用层协议识别的噪声数据,将剩余数据转换成符合神经网络输入的格式,并进行协议标签标注。在数据集划分步骤,把数据划分为训练集和测试集,服务于后续的训练和测试。

在模型训练阶段,由模型提取训练集中数据的特征,根据分类器的判别结果与实际标签的差异,对模型参数进行优化,通过多轮迭代训练出成熟的识别模型。在模型测试阶段,将测试数据输入协议识别模型,通过模型的输出与实际标签进行比对,评估模型的识别效果。下面对协议识别过程中各阶段的关键工作进行介绍。

2.2 数据预处理

数据预处理的目的是从原始网络流量数据中提取出协议数据,并转化成便于分析的格式。本文方法的数据预处理具体可分为四个子步骤:首先是数据清洗,过滤不含应用层载荷的数据包。其次是网络流重组和切分,将同属一个流的数据包进行聚合,并提取固定长度的数据作为后续处理的输入。第三个子步骤是数据归一化,将数据映射到同一范围,以消除量纲不同对后续识别的影响。最后是数据标注,将数据与对应的协议标签进行关联。

第一个子步骤是数据清洗,就是将收集到的网络流量中与应用层协议识别无关的数据包进行过滤。首先,根据数据链路层数据帧头中的FrameType字段判断该数据包是否属于IP包,如果非IP包,则将其过滤。接着,根据IP层Protocol字段确定该数据包是否是UDP数据包或TCP数据包,将不相关数据包(如ICMP数据包)过滤。

第二个子步骤是对网络流进行重组和切分。本文方法关注的是会话级的网络交互,可以是一次完整UDP交互或是一次TCP连接由建立到释放的过程。UDP流没有明显的状态信息,难以准确标识UDP流的开始和结束,可以采用计时器进行判断。如果在预设的时间窗口内没有捕获通信双方之间的下一个UDP数据包,就认为UDP流已经结束。将捕获到的UDP数据包按照顺序拼接,作为一次完整的UDP流。TCP连接可以根据TCP SYN和TCP FIN标志位进行判断,并依据头首部的序列号,将捕获的数据包按序拼接。根据数据包发送方向,可将流量分成正向流和反向流。本文将一次完整交互中的数据包根据发送方向按序拼接,一次交互中的数据包被分成两个方向的流,每个流中含有相同方向数据包的应用层载荷。因为正向流和反向流功能不同,报文格式和内容存在差别,将不同方向的载荷拼接在一起容易混淆特征。以HTTP协议为例,请求报文和响应报文的报文首部有明显差异,报文主体信息也存在较大差别。如果不考虑方向的差异,将所有数据包拼接,杂乱的信息会给后续模型学习造成困难。按照方向对数据包进行重组,可以保证数据流中包含的特征具有较好的区分度,能反映出同方向数据包共同的功能特性,有利于模型训练。

此后,对重组后的数据进行切分。本文方法提取每条数据流前 n 个字节用于协议识别。如果长度不足 n ,末尾用零填充;如果长度超过 n ,丢弃剩余字节。本文采用截取每条数据流前部固定字节的方式进行协议识别。这是因为协议首部字段对于协议识别起到关键作用,但确定协议首部边界往往比较困难。截取前部 n 个字节的数据,一方面可以确保将首部信息包含在内,提高协议识别的准确性。另一方面,网络流中位于前部的数据对于协议类型具有更好的表征作用。同时,由于深度学习模型对输入格式有严格要求,需要对重组后的数据进行切分,提取固定长度的信息以方便整合成符合神经网络输入的形式。本文参考文献[9],提取 $n=784$ 字节的协议数据,并将每个字节映射成0到255之间的十进制数,得到长度为784的一维向量。

第三个子步骤是数据归一化。上一步骤输出的一维向量中每个分量均为十进制数。十进制形式的数值对于模型的收敛而言是不利的。模型的训练速度由梯度下降算法决定,分量值较大时对应反向传播的梯度值也会较大,较大的梯度值会使得模型收敛速度慢甚至不收敛。为了避免此类问题,需要将数据归一到同一量纲,促使收敛加速,提高模型的训练精度。本文方法将每个分量值除以256,转化为[0,1)区间的数值,实现归一化。为了符合协议特征提取模块对输入的严格限制,需要将经过归一化处理的一维向量转换为协议位图,即二维矩阵的形式。具体方法是将一维向量中的每28个元素放置在一行,共28行,形成一个28×28的二维矩阵。

第四个子步骤是对协议数据进行标注。本文方法采用独热编码(one-hot encoding)进行标注,通过 N 位寄存器对 N 种不同协议类别进行编码。举例来看,长度为4的类别列表[HTTP,DNS,POP,SSH]对应4位寄存器,HTTP协议对应的标签为[1,0,0,0],DNS协议对应的标签为[0,1,0,0],以此类推。

2.3　识别模型的工作原理

本文的协议识别模型主要包括特征提取模块和协议分类模块，特征提取模块又细分为空间特征提取模块和时间特征提取模块两部分。模型的总体框架如图2所示。在空间特征提取模块和时间特征提取模块之间有一个嵌入层，嵌入层负责将协议数据进行分片处理并进行位置编码，便于时间特征提取模块通过全局关联进行特征学习。协议分类模块的主要作用是接收提取出的特征信息，对协议所属类别进行判断。

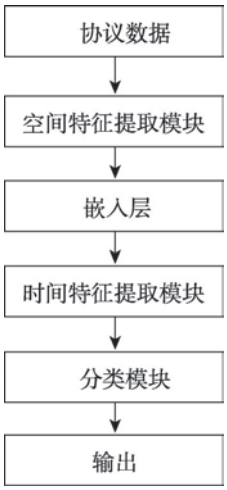

图2　协议识别模型总体框架

Fig.2　General framework of protocol recognition

2.3.1　协议特征提取

(1)空间特征提取子模块

在进行协议特征提取时，首先是提取数据的空间特征。空间特征提取子模块基于SENet和ResNet设计。SENet通过结构化方式，向模型中插入SE块实现通道注意力，能够使模型具备通道级的学习能力。ResNet的基础单元是基础块，ResNet的残差结构可以增强信息在各基础块之间的流动性，在前向传播的过程中能够充分利用特征。在ResNet基础块的卷积层和残差计算之间插入SE块，一方面可以基于卷积通道中信息的关键程度进行特征学习，另一方面可以对关键特征进行再利用，保证模型对特征的充分学习[14]。基于上述分析，采用ResNet作为SENet的载体能够发挥出SENet的优势。ResNet基础块和SE_ResNet基础块结构对比如图3所示。

SENet的核心是SE块，SE块主要通过Squeeze、Excitation和Scale三类操作实现通道注意力计算。Squeeze操作将特征图中每个通道的二维特征压缩为

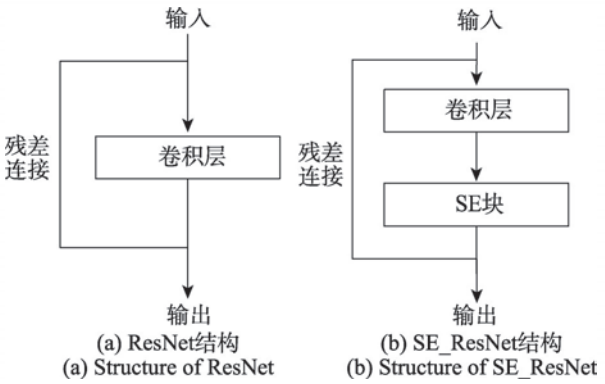

图3　ResNet和SE_ResNet对比

Fig.3　Comparison of ResNet and SE_ResNet

一个通道描述符。Excitation操作根据通道描述符为每个通道计算权重值，权重值越大表示对应通道越重要。Scale操作将权重值和原始特征相乘，使得特征带有通道级别的权重信息。

图4为本文SE_ResNet基础块的具体结构。在SE块的处理过程中，线性层Linear_1和Linear_2分别对每个通道的权重进行分配，并采用SELU(scaled exponential linear unit)函数进行激活。相比于ReLU(rectified linear unit)函数，SELU函数对负值的处理更加稳定，提高了神经元的利用率，更合理地对权重进行再分配。传统做法采用ReLU函数进行激活时，当输入为负时，ReLU函数的输出为0，导致模型对负值通道单元的关注程度没有差别，激活值被截断，神经元无法被激活。而SELU函数的激活值均值和方差保持稳定，当输入小于0时，也能得到非0

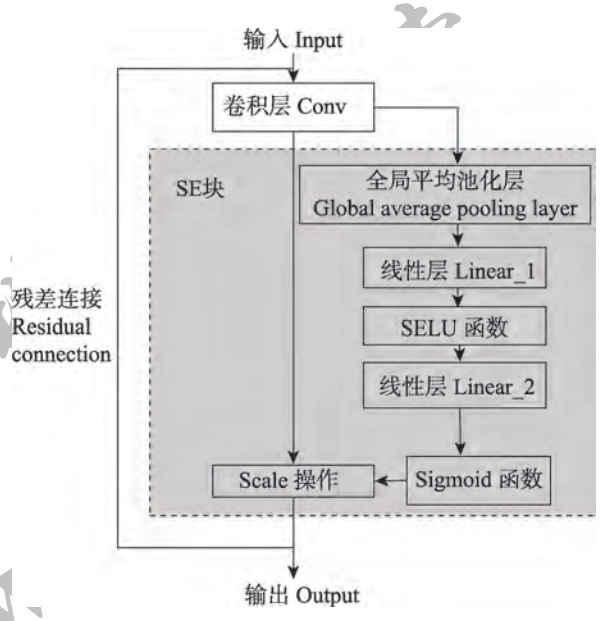

图4　SE_ResNet基础块结构

Fig.4　Structure of SE_ResNet basic block

的输出作为激活值。这样,输入特征值为负的通道单元也将获得非0的激活值作为打分,保证了相应神经元能够被激活,使模型对这些通道的关注程度产生差别。

空间特征提取过程注重特征的充分性。所谓充分性就是模型在尽可能保留数据原始特征的情况下获取关键特征。特征提取主要通过多个基础块的堆叠实现。在残差计算前加入SE块,一方面残差计算让初始特征得以保留的同时与通道级的关键特征进行融合。另一方面,采用基础块堆叠的方式构建子模块,输入某一个基础块的原始特征通过残差连接和提取后的特征进行融合作为下一个基础块的输入,通过多层迭代保证了通道特征得到再次利用,确保了空间特征的充分学习。

空间特征提取子模块的结构如图5所示。特征主要由Layer1、Layer2、Layer3和Layer4递进提取。

Layer1的结构如图6(a)所示。本文中Layer1由两个SE_ResNet基础块构成。Conv1_1表示Layer1中第一个卷积层,SE_block1_1表示Layer1中第一个SE块,后文命名以此类推。SE块的使用使得模型具备对卷积通道关系建模的能力,进而细致提取协议

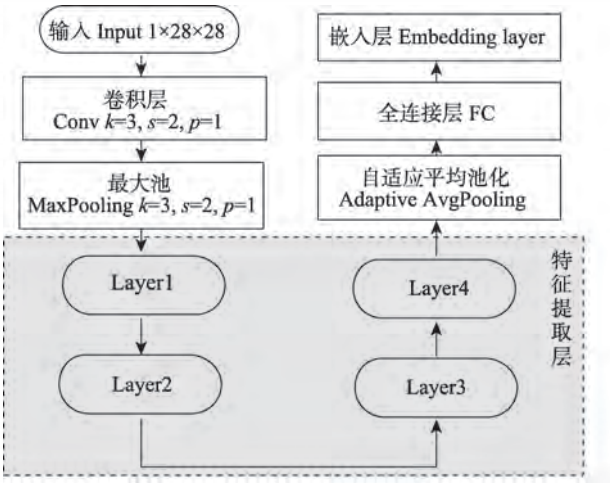

图5 协议空间特征提取子模块结构

Fig.5 Structure of spatial feature extraction sub-module

特征。残差计算保留了输入基础块的原始特征,在前向传播过程中对特征进行重新利用,保证充分学习特征。Layer2结构如图6(b)所示。为了更全面地学习特征,Layer2的第一个基础块在Conv2_1处通过输出通道翻倍实现升维操作。在残差连接处,为了保证输入维度和经过卷积以及通道注意力计算后得到的数据维度一致,采用Conv2_3降采样。Layer3、

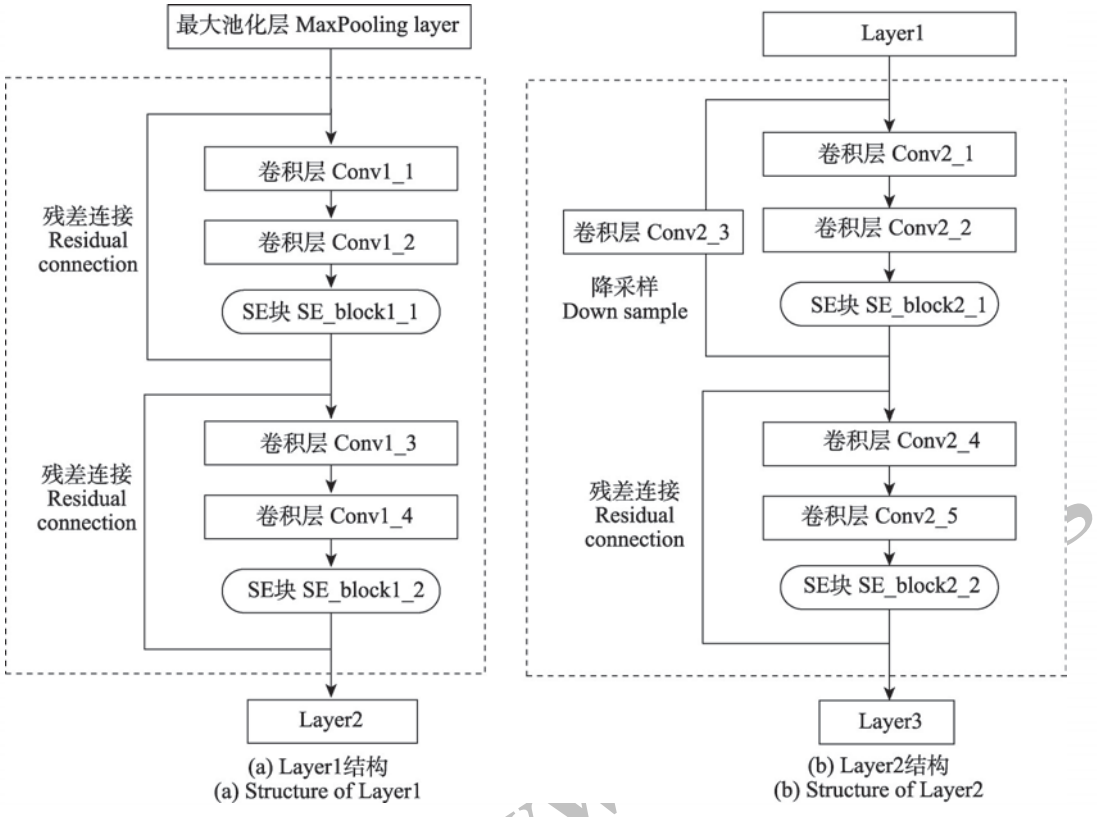

图6 Layer1和Layer2的结构

Fig.6 Structure of Layer1 and Layer2

Layer4与Layer2结构一致。

经过四层提取，原始输入在前向传播中经过了递进式的过滤和再使用，SE_block的使用使模型对关键特征敏感，与残差网络的结合保证了在复杂的空间特征学习过程中既能对原始特征进行相对最大化保留，又能关注到关键特征的融合，充分获取空间特征。

(2)嵌入层

在空间特征提取子模块和时间特征提取子模块之间，设置了一个嵌入层(embedding layer)。嵌入层将提取到的空间特征转换成以片(patch)为单位的序列数据，并将每个片映射到线性空间中，同时进行位置编码，用于表示每个片在初始序列中的位置。以片为单位进行处理主要出于两点考虑：第一，Transformer网络是后续时间特征提取模块的核心，Transformer网络对于输入有严格要求，需要对输入进行序列化处理，并为每个单元进行位置编码以方便利用全局信息。第二，Transformer利用序列中每个元素的位置信息，序列的长度对于Transformer的计算复杂度有一定影响，过长的输入会使注意力计算产生较大的开销。在方法设计上，希望Transformer在有效提取时间特征的同时尽量减小开销。本文方法采用分片的形式，避免了输入序列过长的问题。同时，每个特征片中包含足够多的特征信息，足以保证时间特征学习的需要。经过以上步骤得到时间特征提取子模块的输入，处理流程如图7所示。

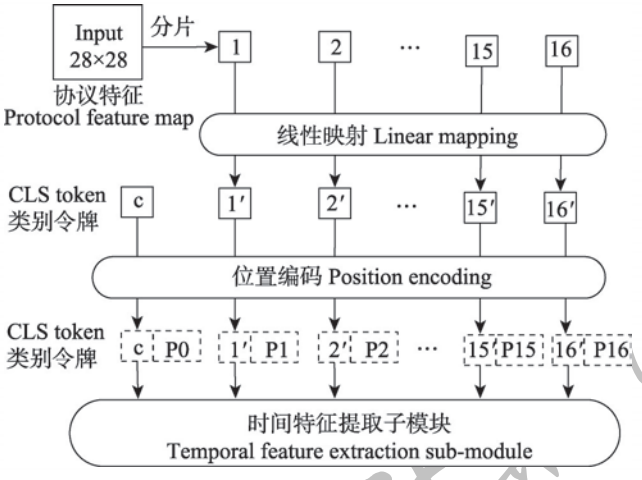

图7　嵌入层的处理

Fig.7　Processing of embedding layer

假设嵌入层需要对大小为 $M\times M$ 的矩阵进行操作，每一片的宽度和高度均为 N，*emb_size* 表示分片大小，为 N^2。*num_patch* 表示分片个数，分片个数为 $\left(\frac{M}{N}\right)^2$，公式如下：

$$emb\_size = N^2 \tag{1}$$

$$num\_patch = \left(\frac{M}{N}\right)^2 \tag{2}$$

本文采用的参数为 $M=28$，$N=7$，$emb\_size=49$，$num\_patch=16$。将28×28的协议空间位图划分成16个大小为7×7的片并进行线性映射。为了保证操作效率，本文通过二维卷积层完成此步骤，卷积层输入通道数为1，输出通道数和 *emb_size* 均为49，卷积核大小和 *emb_size* 均为7×7，步长为7。卷积操作完成后对分片进行展平，最终实现16个协议特征分片的映射，如图7中“1”号到“16”号分片映射为“1”到“16′”。对于类别的判定，引入类别令牌(CLS token)[15]与剩余分片单元进行自注意力的计算作为分类依据。具体做法是，在完成协议数据分片的映射后，在16个分片前加入类别令牌作为最终分类判别位，类别令牌可以视作人为加入用于分类任务的分片(图7中“c”分片)，其大小和其他分片大小保持一致。接着，对包括类别令牌在内的所有分片进行位置编码，本文中位置编码为一个含有17个分量的向量(如图7中“P0”到“P16”)，每个分量值随机生成，最终与映射进行连接完成编码。如图7所示，编码完成后，P1号位置到P16号位置用于时间特征的学习，P0号位置对应类别令牌，类别令牌所产生的输出最终作为数据类别的判断依据。

(3)时间特征提取子模块

如图8所示，本文的时间特征提取子模块基于Transformer编码器，通过自注意力机制实现对整个序列的全局特征表达，避免传统序列模型过度依赖相邻单元的问题，并能够建立长距离依赖。时间特征提取模块由 L 层编码器堆叠而来，本文方法中 L 被设置为2。每个编码器包括多头注意力(multi-head attention)层和多层感知机(multi-layer perceptron，MLP)层。输入经过标准化和维度还原后，多头注意力层计算每个协议分片的表示向量。MLP层通过全连接网络对每个协议分片的表示向量进行非线性变化，并与多头注意力层的表示向量通过残差计算得到协议数据分片的全局表示向量。类别令牌处的输出用于判别协议类型。

多头注意力层的三个输入向量 $\boldsymbol{K}$、$\boldsymbol{Q}$、$\boldsymbol{V}$，均由嵌入层编码后的协议数据分片通过线性映射得到。

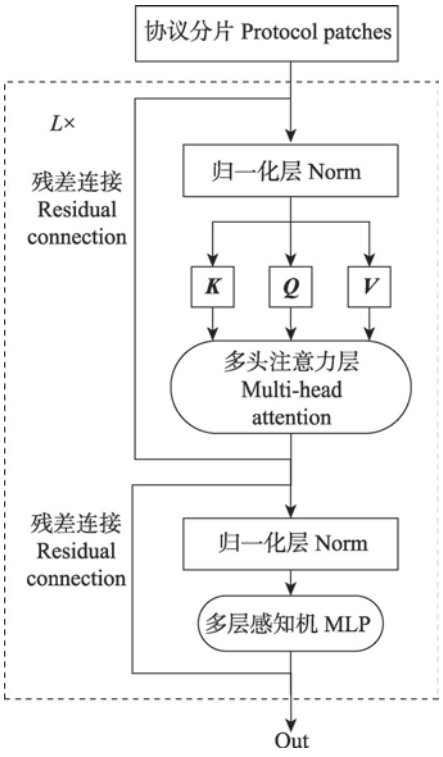

图8 协议时间特征提取子模块的结构

Fig.8 Structure of protocol temporal feature extraction sub-module

$\boldsymbol{K}$ 代表键向量，$\boldsymbol{Q}$ 代表查询向量，二者维度相同，均为 d_K。$\boldsymbol{V}$ 为值向量，其维度为 d_V。自注意力计算可以为每个协议分片计算出一个特征表达，其主要思想是基于 $\boldsymbol{K}$ 和 $\boldsymbol{Q}$ 之间的乘积，采用Softmax函数计算输入的分片序列中每一个协议分片与剩余协议分片的相关系数，再利用这个相关系数对相对应的 $\boldsymbol{V}$ 值进行缩放，其计算如式(3)所示。

$$Attention(\boldsymbol{K},\boldsymbol{Q},\boldsymbol{V})=\mathrm{Softmax}\left(\frac{\boldsymbol{Q}\boldsymbol{K}^{\mathrm{T}}}{\sqrt{d_K}}\right)\boldsymbol{V} \tag{3}$$

如图9所示，对于 h 个注意力头，需要将输入划分成 h 份后进行注意力计算，将每份的结果连接形成最终的输出。假设每层划分后的参数为 $\boldsymbol{W}_i^K$、$\boldsymbol{W}_i^Q$、$\boldsymbol{W}_i^V$。经过注意力运算后相连接，最终与 $\boldsymbol{W}^o$ 相乘，将维度恢复为输入维度。$\boldsymbol{W}_i\boldsymbol{K}$、$\boldsymbol{W}_i\boldsymbol{Q}$ 为 $d_{\mathrm{model}}\times d_K$ 型参数矩阵，$\boldsymbol{W}_i\boldsymbol{V}$ 为 $d_{\mathrm{model}}\times d_V$ 型矩阵。$\boldsymbol{W}^o$ 为 $d_V\times d_{\mathrm{model}}$ 型矩阵，d_{model} 为协议分片大小，$d_V=d_K=\dfrac{d_{\mathrm{model}}}{h}$。多头注意力的计算如式(4)、(5)所示。

$$head_i=Attention(\boldsymbol{K}\boldsymbol{W}_i^K,\boldsymbol{Q}\boldsymbol{W}_i^Q,\boldsymbol{V}\boldsymbol{W}_i^V) \tag{4}$$

$$Multi(\boldsymbol{K},\boldsymbol{Q},\boldsymbol{V})=Concat(head_{i\sim h})\boldsymbol{W}^o \tag{5}$$

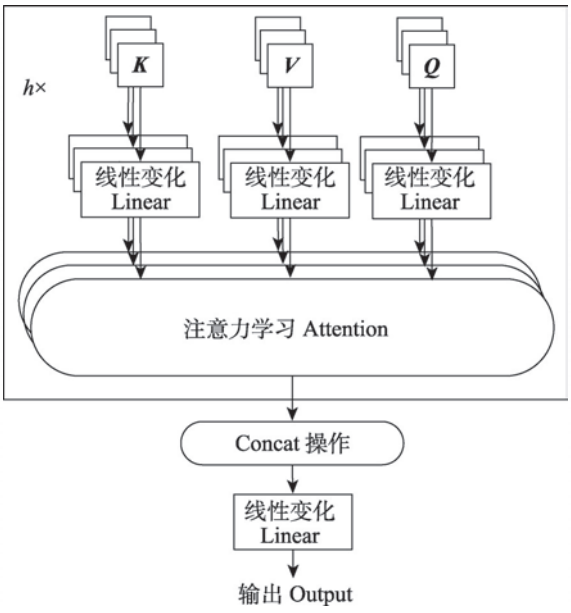

图9 多头注意力机制框架

Fig.9 Framework of multi-head attention

通过多头注意力层的处理后，还需要通过MLP层对时间特征进行进一步整合。MLP多层感知机由两个线性层构成，第一个线性层的输入维度为 d_{model}，通过一个扩张因子 E 对输入进行上采样，输出维度为 $E\times d_{\mathrm{model}}$，接着通过第二个线性层进行维度恢复，进而和上一层输入进行残差连接得到输出后提交给分类模块。本文中的 d_{model} 设置为49。

2.3.2 分类模块

在获得协议数据的时间特征和空间特征之后，将特征信息输入线性层，线性层的输入维度为 d_{model}，输出维度为协议类别数量，可以得到用于区分协议类型的特征向量，由Softmax函数对特征向量进行激活，最终得到协议数据属于不同协议类别的概率，其中概率最大的即为当前协议数据的类别。单类预测概率 y_hat_i 的计算公式如式(6)所示。

$$y\_hat_i=\frac{\exp(\boldsymbol{W}_i\boldsymbol{x})}{\sum_{j=1}^{k}\exp(\boldsymbol{W}_j\boldsymbol{x})} \tag{6}$$

其中，$\boldsymbol{x}$ 为代表特征的列向量，$\boldsymbol{W}_i$ 为当前类别对应的权重向量，k 表示协议分类数。采用 k 位寄存器向量 $\boldsymbol{Y}_{pred}$ 存储每一类的判别概率，其中下标代表所属类别，采用argmax函数选取概率值最大的位置下标作为最终分类结果 $pred$，如式(7)、(8)所示。

$$\boldsymbol{Y}_{pred}=[y\_hat_1,y\_hat_2,\cdots,y\_hat_k] \tag{7}$$

$$pred=\mathrm{argmax}(\boldsymbol{Y}_{pred}) \tag{8}$$

3　实验验证

为了验证本文方法的有效性，进行了实验。实验中主要考虑到如下几方面因素：

（1）SE块的应用与否对模型性能的影响；

（2）不同时间提取子模块结构对模型性能的影响；

（3）与目前主流的协议识别模型的横向对比。

本文采用混淆矩阵辅助计算评估指标，将准确率（accuracy）、召回率（recall）、*F*1值和精确率（precision）作为协议识别模型的度量指标。计算公式如式（9）~（12）所示。

$$accuracy = \frac{TP + TN}{TP + TN + FP + FN} \tag{9}$$

$$precision = \frac{TP}{TP + FP} \tag{10}$$

$$recall = \frac{TP}{TP + FN} \tag{11}$$

$$F1\_Score = \frac{2 \times precision \times recall}{precision + recall} \tag{12}$$

其中，*TP*（true positive）表示将正类预测为正类数；*FN*（false negative）表示将正类预测为负类数；*FP*（false positive）表示将负类预测为正类数；*TN*（true negative）表示将负类预测为负类数。

3.1　实验环境和数据

本文实验环境部署在Windows 10操作系统中，CPU为AMD Ryzen 5 4600H，GPU为NVIDIA GeForce GTX 1650Ti，专用GPU内存为4 GB。环境基于Python3.8进行搭建，CUDA版本为11.6，深度学习框架采用Pytorch，版本为1.12.1。

本文实验数据采用ISXC2012和CSE_CIC_IDS2018数据集混合的方式构建。ISXC2012数据集涵盖近1 512 000个数据包，包含了为期7天的流量数据[16]，目前在协议识别、入侵监测等领域仍广泛用作基准数据集。CSE_CIC_IDS2018数据集包含为期5天的网络流量数据，涵盖种类众多的数据包。该数据集提供了PCAP（packet capture）格式文件，可以帮助分析者在此基础上对流量进行分析[17]。采用两个代表性公开数据集混合的方式一方面可以保证数据的多样性，另一方面采用复杂数据集进行训练可以提高模型的泛化性。本文从两个数据集中分别提取会话进行流重组，重组后的流中包含数据包的应用层载荷。将两组公开数据集合并后，从中抽取了6类应用层协议，实验中6类协议数据的分布如表1所示。

协议数据标签采用独热编码的形式。举例来看，DNS协议对应编号为0，则将初始向量下标为0的分量置1，最终得到DNS协议的独热标签为[1,0,0,0,0,0]。

表1　协议数据分布

Table 1　Distribution of protocol data

编号	协议类别	数量	占比/%
1	DNS	10 000	16.67
2	FTP	10 000	16.67
3	HTTP	10 000	16.67
4	IMAP	10 000	16.67
5	SSH	10 000	16.67
6	POP	10 000	16.67
	总数	60 000	100.00

本文将处理好的协议数据根据索引进一步打乱，选取前3/4作为训练集，剩余1/4数据作为测试集。

3.2　SE块的应用与否对模型性能的影响

协议空间特征是协议数据的直观表达，空间特征的学习对于协议识别具有重要意义，直接影响到最终判别结果。本文在空间提取结构中加入SE块，使得模型能够根据卷积通道之间的关系为重要通道赋予高权重。为了验证SE块的有效性，设计对照实验，在相同时间特征提取结构的基础上，测试架构中SE块的增加与否对模型性能的影响，选用的评价指标包括准确率、精确率、*F*1值和训练时间。空间特征提取模块主要结构参数如表2所示，实验结果如表3所示。

从实验结果来看，在协议识别模型中加入SE块可以使得模型能够更加充分地获取协议数据的特征，从而有效地提升协议识别的效果。但模型对协议数据进行更加细致特征提取的同时，会带来一定的过程开销，因此训练时间有所增加。

如图10所示，SE块的加入使每类协议的识别效果均有所提升，其中DNS协议识别率达到了100%，但对于HTTP协议而言，虽然识别率有所提升，但仍然相对偏低。分析其原因，HTTP协议的请求和响应数据包含有不同的结构化信息，同时数据格式具有灵活性。为了支持不同的功能，HTTP协议设计得相对复杂，造成特征不稳定。即使通过对用户的个性数据进行模糊化降低了用户信息对协议识别的影响，但仍无法很好地避免不稳定特征对模型识别效果的干扰。另外，对于SSH协议为代表的加密协议，加密算法的运用使得协议特征遭到混淆，因此模型处理中的细化处理对该类协议识别效果提升有限。

表2 空间特征提取模块关键结构的参数

Table 2 Parameters of key structures of spatial feature extraction module

结构	输入	输出	卷积核	步长	填充
Conv1_1	64	64	3	1	1
Conv1_2	64	64	3	1	1
Conv1_3	64	64	3	1	1
Conv1_4	64	64	3	1	1
Conv2_1	64	128	3	2	1
Conv2_2	128	128	3	1	1
Conv2_3	64	128	1	2	0
Conv2_4	128	128	3	1	1
Conv2_5	128	128	3	1	1
Conv3_1	128	256	3	2	0
Conv3_2	256	256	3	1	1
Conv3_3	128	256	1	2	0
Conv3_4	256	256	3	1	1
Conv3_5	256	256	3	1	1
Conv4_1	256	512	3	2	0
Conv4_2	512	512	3	1	1
Conv4_3	256	512	1	2	0
Conv4_4	512	512	3	1	1
Conv4_5	512	512	3	1	1
SE_block1_1	64	64	—	—	—
SE_block1_2	64	64	—	—	—
SE_block2_1	128	128	—	—	—
SE_block2_2	128	128	—	—	—
SE_block3_1	256	256	—	—	—
SE_block3_2	256	256	—	—	—
SE_block4_1	512	512	—	—	—
SE_block4_2	512	512	—	—	—

表3 SE块对模型效果的影响

Table 3 Influence of SE block on model

是否加入SE块	总体准确率/%	精确率/%	*F*1值/%	轮次	训练时间/s
是	**99.20**	**99.03**	**98.99**	**20**	**439.3**
否	99.00	98.80	98.76	20	421.3

3.3 时间提取子模块结构对模型性能的影响

本文的时间特征提取子模块基于Transformer进行设计，采用编码器构建时间特征提取层对序列化的协议数据进行特征提取。完整的时间特征提取子模块可由 L 层时间特征提取层进行堆叠形成。L 的取值直接影响到模型的时间耗费和协议时间特征的提取效果，过少的堆叠可能会造成特征提取不足，过分的堆叠可能造成不必要的资源消耗。

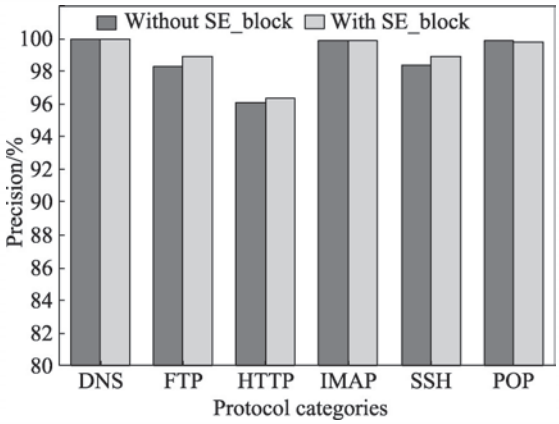

图10 SE块对精确率的影响

Fig.10 Influence of SE block on precision

本文基于 L 值的不同设计了六种不同的时间特征提取子模块，分析 L 的取值对模型识别效果的影响，实验结果如表4所示。

表4 时间特征提取模块层数量的影响

Table 4 Influence of different number of temporal feature extraction layers

实验编号	时间特征提取层数量 L	总体准确率/%	轮次	训练时间/s
1	1	99.08	20	411.3
2	2	**99.20**	**20**	**439.3**
3	3	98.79	20	464.5
4	4	98.91	20	490.6
5	5	99.13	20	515.3
6	6	98.90	20	551.5

当 L 取值为1时，单层结构的学习在协议时间特征的提取方面缺乏充分性，从而导致模型的较低识别率。当 L 增加至2时，增加的时间特征提取层能够更加全面地提取时间特征，模型的识别效果得到提高。然而，当 L 大于等于3时，时间特征提取子模块的复杂性过高，导致模型提取的特征过度，造成模型性能的退化。

特征学习基于Transformer架构，使用随机梯度下降（stochastic gradient descent，SGD）函数作为优化函数对于资源消耗相对较大，并存在参数选择困难、模型收敛困难等问题[18]。如表5所示，选用SGD函数作为优化函数，学习率设置为0.001的情况下，PrtCNN[9]达到收敛条件需要415.3 s，训练轮次为150轮。同等条件下，本文模型收敛较慢，需要超50轮的训练，每轮训练时间明显超过PrtCNN模型的每轮训练时间，总训练时间超过900 s。

表5　SGD函数优化策略下不同模型的收敛时间

Table 5　Convergence time of different models under SGD function optimization strategy

实验编号	模型	轮次	训练时间/s
1	本文模型	**50**	**1 024.5**
2	PrtCNN	150	415.3

本文使用AdamW（Adam weight decay regularization）优化器对模型进行优化。如表6所示，相较于SGD优化器，AdamW优化器在参数量较大的情况下具有更高的运算效率，能够节约模型训练时间，并在相对较少的训练轮次下达到更高的识别准确率。本文初始学习率设置为0.001，betas设定为（0.90，0.99），权重衰减设置为0.05。到达收敛条件需要439.3 s。相同学习率设置下，相比于使用SGD函数，时间耗费明显降低。如图11、图12所示，相比于SGD优化函数，使用AdamW函数可以在较少轮次下，取得较好的协议识别结果。

表6　不同优化策略下本文模型的收敛时间

Table 6　Convergence time of this paper model under different optimization strategies

编号	优化函数	轮次	训练时间/s
1	SGD	50	1 024.5
2	AdamW	**20**	**439.3**

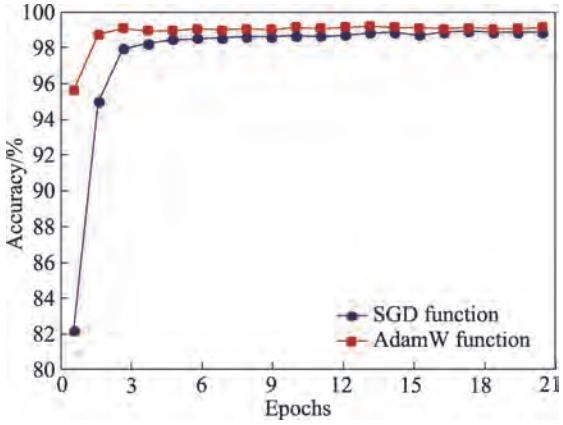

图11　两种策略下准确率曲线的对比

Fig.11　Comparison of accuracy curves under two strategies

3.4　与其他协议识别模型做效果对比

为了进行横向对比，实验中包含了文献[9]、文献[11]和文献[13]提到的方法和本文方法。文献[9]提出的PrtCNN使用二维卷积网络进行特征提取，利用空间特征对协议进行识别，在基于空间特征的识别方法中具有代表性。文献[11]中Wu等人的方法，首先利用改进的ResNet通过添加分支的方式提取空间特征，再经过BiGRU的处理抑制时间特征提取中长距离造成的影响。文献[13]所提STF-SA模型采用LeNet-5和Transformer结合的方式对协议的时空特征进行提取，该方法虽然以加密协议作为主要研究对象，但也适用于通用协议的识别。各类模型识别结果如表7、表8和图13所示。

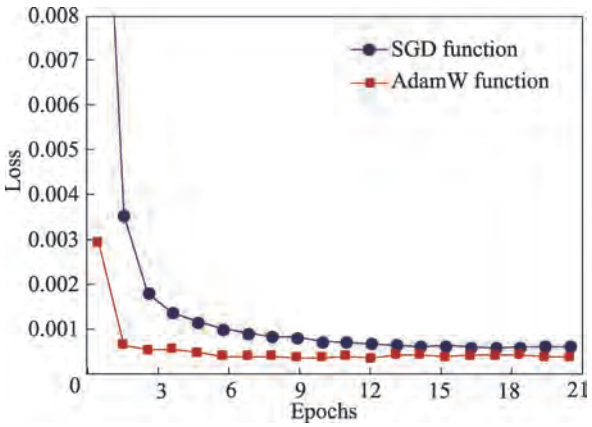

图12　两种策略下损失曲线的对比

Fig.12　Comparison of loss curves under two strategies

根据表7和图13可以观察到本文模型的总体准

表7　不同模型协议识别性能

Table 7　Protocol recognition performance of different models　单位：%

模型	准确率	*F*1值	召回率
本文模型	**99.20**	**98.99**	**98.96**
PrtCNN	97.13	97.25	97.19
Method_of_Wu	95.41	95.54	96.68
STF-SA	98.87	98.42	98.46

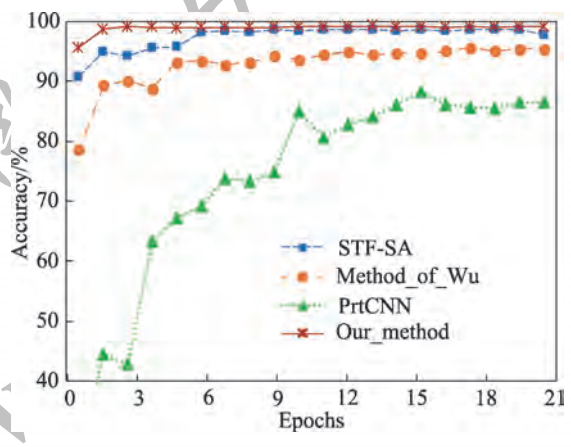

图13　各模型总体准确率

Fig.13　Protocol recognition accuracy of different models

确率、*F*1值、召回率均高于对比模型。PrtCNN仅从单一的空间维度对协议特征进行学习且特征学习中没有侧重,特征提取不充分。本文模型从空间和时间两个维度出发,且空间特征提取更加充分细致,最终得到的用于分类的特征中既包含了空间信息也包含了时间信息,使模型性能整体得到提高。对比Wu等人的方法,在空间特征提取阶段,本文向网络中加入了SE块,使模型在此阶段能够捕捉到卷积通道中的活跃特征;在时间特征提取阶段,本文方法采用了Transformer的编码器结构,解决了时间特征提取过程中距离依赖的问题,最终得到的融合特征相比Wu等人的方法更加充分和全面。相比于STF-SA模型,本文空间特征提取基于残差网络和SENet,一方面关注到关键特征的再学习以提高充分性,另一方面关注空间特征学习中的关键特征的获取,向残差结构中加入SE块。时间特征提取方面,将协议特征进行分片而不是基于大量序列,降低了时间特征学习的成本,同时分片中含有的信息也更加充足,从而使得本文模型整体优于STF-SA模型。

四种模型的识别精确度如表8所示,本文模型能够有效地提升对于DNS、FTP、IMAP和POP协议的识别精确率。分析其原因,一部分在于SE块的引入使得模型可以通过通道注意力学习更加细节的特征,另一部分由于文本协议的特性,采用时间特征学习能进一步提取有效表征数据。但是模型仍存在一些局限。针对HTTP协议,由于其特征不稳定,识别效果虽相比于部分模型有所提高,但仍需改进,通过选取更好的表征数据以提高识别效果。对于SSH这类采用加密方法对应用层数据特征进行模糊化处理的协议而言,本文方法处理过程中关注细节特征,因此不利于此类加密协议的识别。

表8 不同模型协议识别精确性

Table 8 Protocol recognition precision of different models 单位:%

模型	DNS	FTP	HTTP	IMAP	SSH	POP
本文模型	**100.00**	**98.88**	**96.31**	**99.87**	**98.92**	**99.75**
PrtCNN	99.91	97.95	94.93	91.26	100.00	99.11
Method_of_Wu	100.00	98.54	80.89	99.83	95.37	99.47
STF-SA	99.91	97.42	98.89	99.67	95.54	99.35

4 结束语

针对现有方法对协议特征提取不充分、不全面的问题,本文提出了基于SENet和Transformer的协议识别方法。在空间特征提取阶段通过向残差网络中加入SE块,使模型在获取通道级细节特征的同时充分学习。在时间特征提取阶段基于Transformer的编码器进行时间特征的学习,从全局的角度学习时间特征以保证时间特征提取的全面性。与本领域中其他模型的横向对比实验表明,本文模型的协议识别性能高于对比模型。本文模型的主要问题是对HTTP协议和加密协议的识别效果不理想。另外,对于先前未知的协议,需要收集相应的协议数据构建训练集对模型进行训练。相对复杂的模型结构与庞大的参数量意味着模型需要更多的计算资源和时间,消耗较高。未来将进一步研究HTTP协议特征,提取更具代表性的信息来提升协议识别的效果。基于加密协议的特点,针对加密协议的识别问题,将进行更深入的研究。同时,将进一步优化训练算法,提高模型的训练效率。

参考文献:

[1] 冯文博, 洪征, 吴礼发, 等. 网络协议识别技术综述[J]. 计算机应用, 2019, 39(12): 3604-3614.
FENG W B, HONG Z, WU L F, et al. Review of network protocol recognition techniques[J]. Journal of Computer Applications, 2019, 39(12): 3604-3614.

[2] XU W, ZOU F. Obfuscated Tor traffic identification based on sliding window[J]. Security and Communication Networks, 2021. DOI:10.1155/2021/5587837.

[3] NASIR M, JAVED A R, TARIQ M A, et al. Feature engineering and deep learning-based intrusion detection framework for securing edge IoT[J]. The Journal of Supercomputing, 2022, 78: 8852-8866.

[4] HE K, ZHANG X, REN S, et al. Deep residual learning for image recognition[C]//Proceedings of the 2016 IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas, Jun 27-30, 2016. Piscataway: IEEE, 2016: 770-778.

[5] NIU Z, ZHONG G, YU H. A review on the attention mechanism of deep learning[J]. Neurocomputing, 2021, 452: 48-62.

[6] HU J, SHEN L, SUN G. Squeeze-and-excitation networks [C]//Proceedings of the 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition, Salt Lake City, Jun 18-22, 2018. Piscataway: IEEE, 2018: 7132-7141.

[7] REN F, JIANG Z, LIU J. A bi-directional LSTM model with attention for malicious URL detection[C]//Proceedings of the 2019 IEEE 4th Advanced Information Technology, Electronic and Automation Control Conference, Chengdu, 2019. Piscataway: IEEE, 2019: 300-305.

[8] VASWANI A, SHAZEER N, PARMAR N, et al. Attention

is all you need[C]//Advances in Neural Information Processing Systems 30, Long Beach, Dec 4-9, 2017: 5998-6008.

[9] FENG W, HONG Z, WU L, et al. Network protocol recognition based on convolutional neural network[J]. China Communications, 2020, 17(4): 125-139.

[10] WEI W, GU H, DENG W, et al. ABL-TC: a lightweight design for network traffic classification empowered by deep learning[J]. Neurocomputing, 2022, 489: 333-344.

[11] 吴吉胜, 洪征, 马甜甜, 等. 基于残差网络和循环神经网络混合模型的应用层协议识别方法[J]. 计算机科学, 2022, 49(11): 293-301.
WU J S, HONG Z, MA T T, et al. Application layer protocol recognition based on residual network and recurrent neural network[J]. Computer Science, 2020, 49(11): 293-301.

[12] SARHANGIAN F, KASHEF R, JASEEMUDDIN M. Efficient traffic classification using hybrid deep learning[C]// Proceedings of the 2021 IEEE International Systems Conference, Vancouver, 2021. Piscataway: IEEE, 2021: 1-8.

[13] 彭瑶. 基于深度学习的加密流量分类方法研究[D]. 广州: 广州大学, 2022.
PENG Y. Research on encrypted traffic classification method based on deep learning[D]. Guangzhou: Guangzhou University, 2022.

[14] HE K, ZHANG X, REN S, et al. Identity mappings in deep residual networks[C]//Proceedings of the 14th European Conference on Computer Vision, Amsterdam, Oct 11-14, 2016. Cham: Springer, 2016: 630-645.

[15] DOSOVITSKIY A, BEYER L, KOLESNIKOV A, et al. An image is worth 16x16 words: Transformers for image recognition at scale[J]. arXiv:2010.11929, 2020.

[16] GHANEM W A H M, GHALEB S A A, JANTAN A, et al. Cyber intrusion detection system based on a multiobjective binary bat algorithm for feature selection and enhanced bat algorithm for parameter optimization in neural networks [J]. IEEE Access, 2022, 10: 76318-76339.

[17] LIU L, ENGELEN G, LYNAR T, et al. Error prevalence in NIDS datasets: a case study on CIC-IDS-2017 and CSE-CIC-IDS-2018[C]//Proceedings of the 2022 IEEE Conference on Communications and Network Security, Austin, Oct 3-5, 2022. Piscataway: IEEE, 2022: 254-262.

[18] DE S, GOLDSTEIN T. Efficient distributed SGD with variance reduction[C]//Proceedings of the 2016 IEEE 16th International Conference on Data Mining, Barcelona, Dec 12-15, 2016. Piscataway: IEEE, 2016: 111-120.

陈乾(1999—),男,江苏南京人,硕士研究生,主要研究方向为网络空间安全、协议逆向工程。
CHEN Qian, born in 1999, M.S. candidate. His research interests include cyberspace security and protocol reverse engineering.

洪征(1979—),男,江苏南京人,博士,副教授,主要研究方向为网络空间安全、协议逆向工程。
HONG Zheng, born in 1979, Ph.D., associate professor. His research interests include cyberspace security and protocol reverse engineering.

司健鹏(1996—),男,河北邢台人,硕士研究生,主要研究方向为软件安全、协议逆向工程。
SI Jianpeng, born in 1996, M.S. candidate. His research interests include software security and protocol reverse engineering.