DOI：10.13718/j.cnki.xdzk.2021.12.023

基于双域注意力和元学习的移动应用行为识别

张文君，　陈丹伟

南京邮电大学 计算机学院、软件学院、网络空间安全学院，南京 210000

摘要：移动智能设备和移动应用承载了诸多个人信息和办公娱乐功能，通过分析移动应用在使用时产生的网络流量，可以在网络管理、隐私保护以及行为识别方面提供有价值的信息. 文章设计了一种基于双域注意力机制和元学习的识别模型，首先，通过深度可分离卷积模块进行特征提取；其次，通过注意力机制模块从通道和空间 2 个维度提取注意力，增强行为识别样本的图像纹理特征；同时，利用元学习的策略，进行多任务学习，使得模型在面对新的小样本识别任务时可以有更快速高效的识别效果. 实验结果表明，相比于其他的小样本识别模型，本文模型能够有效地识别出移动应用行为特征.

关　键　词：移动应用行为识别；深度学习；注意力机制；元学习；深度可分离卷积；小样本分类

中图分类号：TP391.1　　**文献标志码**：A

开放科学(资源服务)标识码(OSID)：

文 章 编 号：1673－9868(2021)12－0198－11

Mobile Application Behavior Recognition Based on Dual-Domain Attention and Meta-Learning

ZHANG Wenjun，　CHEN Danwei

School of Computer / School of Software / School of Cyberspace Security of Nanjing University of Posts and Telecommunications，Nanjing 210000，China

Abstract：Mobile smart devices and mobile applications carry a lot of personal information and office entertainment functions. Analysis of the network traffic generated when mobile applications are used can provide valuable information in terms of network management，privacy protection and behavior recognition. In this paper，a recognition model based on dual-domain attention mechanism and meta-learning is designed. First，feature extraction is performed through the deep separable convolution module. Then，attention is extracted from the channel and space dimensions through the attention mechanism module to en-

收稿日期：2021－02－08
基金项目：国家重点研发计划项目(2019YFB2101704).
作者简介：张文君，硕士生，主要从事网络流量分析方面的研究.

hance the texture features of the behavior recognition samples. At the same time, the meta-learning strategy is used to perform multi-task learning, so that the model can have a faster and more efficient recognition effect when facing new small-sample recognition tasks. The results of an experiment show that compared with other small sample recognition models, the model described in this paper can more effectively recognize mobile application behaviors.

Key words: mobile application behavior recognition; deep learning; attention mechanism; meta-learning; deep separable convolution; small sample classification

随着移动设备智能化程度的提高，手机承载了诸多以前只能通过电脑实现的功能和任务. 与此同时，5G 已经渐渐进入我们的日常生活，2015－2020 年中国手机即时通信用户规模逐年增长. 事实上，网络性能的大幅提升不仅为即时通信类别的应用带来了较大改变，同时也使得移动设备能够为生活、工作、娱乐等各方面的应用需求提供支持. 由此可见，5G 对移动端应用市场的蓬勃发展起到了极大的促进作用.

用户在使用每个移动端的应用时会产生各种各样的网络流量，通过分析这些流量可以获取很多信息. 比如，第一，可以对用户的行为作出分析，或者对某个地区某个年龄的用户行为作出分析，从而刻画出用户形象以便更好地推荐；第二，可以实现从攻击者的视角尽早发现一些有恶性行为的应用并进行防范，从而避免出现隐私泄露等安全问题；第三，可以满足某些场景或企业的个性化需求，提高网络服务质量.

传统的移动应用流量识别主要有 3 种方式，基于端口号识别、基于 DPI 深度包检测识别和基于机器学习的方式. 其中，基于端口号和 DPI 深度包检测的流量识别方法是依靠研究人员制订的规则来进行匹配和识别的，此外，基于统计和基于行为的方法都属于传统的机器学习方法，故仍需要手工进行特征选择后，模型才能依据既定特征对待识别样本进行识别. 然而，随着近几年深度学习在各领域的大放异彩，研究学者开始尝试用深度学习解决传统流量识别方法中的问题. 图 1 展示了不同流量识别方法的具体流程，深色块表示机器完成的部分. 可以看出，使用深度学习方式可以省略人工进行特征设计的步骤，这已在图像分类、自然语言处理等多个领域得到了验证. 鉴于此，在流量识别领域我们同样能够借助深度学习来提高流量识别的能力. 同时，考虑到在实际应用场景中移动应用迭代频繁的情况，本文采用了元学习多任务训练的方式，解决了识别模型冷启动的问题，实现了小样本场景下的移动应用行为识别.

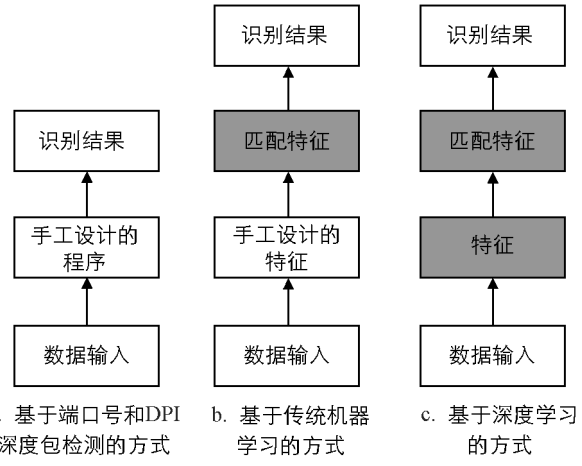

图 1　不同流量识别方式的流程

1　国内外相关研究工作

网络流量分析[1-3]一直是网络安全领域一个重要的研究方向. 如今人们对手机的依赖程度越来越高，手机承载着我们越来越多的隐私信息，例如，照片、定位信息、文件甚至是各类金融信息，因此我们尝试通过对移动应用使用中产生的网络流量进行分析，识别用户的行为[4]，来达到分析用户的行为模式或者是用于场景化管理[5-6]的目的. 网络流量分析一般分为流量采集、流量处理、特征分析、结果评估 4 个步骤.

近年来越来越多研究人员开始利用网络流量来识别移动应用中用户的行为，Coull 等人[6]在研究

iMessage用户产生的流量时，尽管iMessage对流量进行了加密处理，但通过分析用户与苹果服务器交互时产生的网络数据包的大小等侧面信息，仍可以获取用户的相关操作特征，例如可以识别出消息的长度、语言的种类等，同时，对发消息的行为，包括输入状态和阅读状态等5种行为进行了区分，正确率在90%以上. Lee等人[7]研究了韩国的一款通信移动应用，在对用户发消息、发图片、添加好友等11种行为产生的网络流量差异进行分析时，通过提取这些加密数据流的数据报文也能够对用户的行为进行识别. Li等人[8]提出在进行移动应用行为识别时，选取数据报文的时间序列和长度序列作为特征进行分析，也可以不受加密的影响.

上述文献使用的都是传统的分步策略，目前深度学习已经渐渐开始取代手工设计特征[9]的操作过程. Nan等人[10]提出将原始的网络数据流量直接作为输入，使用一种基于栈式自编码器SAE的识别方法，开创了端到端方法识别流量的先河. 王伟[11]同样是将原始的网络数据流量直接作为输入，结合卷积神经网络来输出模型的分类结果，将网络流量的识别很好地与深度学习方式结合到了一起，并且取得了很好的实验效果. 因此，采用深度学习的方式，对网络流量的原始数据进行学习在流量识别领域是可行且效果可观的.

2 基于双域注意力和元学习的移动应用行为识别方法

基于双域注意力机制和元学习的移动应用行为识别分类模型如图2所示，在对流量数据进行可视化操作后，利用深度可分离卷积、双域注意力和元学习训练策略[12]等模块来完成在小样本情况下的移动应用行为识别任务. 其中，用深度可分离卷积取代常规卷积操作，是为了在确保识别准确率的同时降低模型参数量，实现更好的泛化性；使用双域注意力模块，可以从通道域和空间域2个方面提高输入特征图中有用信息的权重，抑制无用信息，提高识别的准确率；使用元学习的训练方式，将双域注意力模型作为行为识别的基础模型，经过大量多任务训练后，能够解决识别模型的冷启动问题，实现在小样本情况时的移动应用行为特征识别.

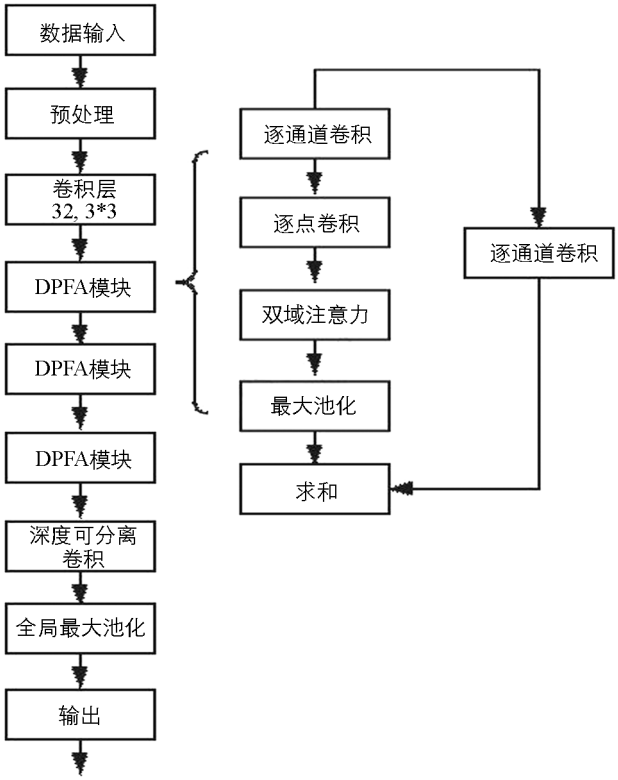

图2 本文的移动应用行为识别模型

2.1 采用深度可分离卷积提取特征

在对训练图片进行预处理后，特征提取部分本文使用深度可分离卷积来代替常规的卷积操作. 采用深度可分离卷积(Depthwise Separable Convolution)的目的是通过减少模型的参数来提高计算的效率.

深度可分离卷积主要由逐通道卷积和逐点卷积2个部分组成. 逐通道卷积(Depthwise Convolution)是作用于通道层面，如图3a所示展示的是逐通道卷积的模型，一个卷积核对应一个通道. 逐点卷积(Pointwise Convolution)与常规的卷积类似，如图3b所示展示的是逐点卷积的模型，其实就是将上一步得到的特

征图在深度上进行加权组合，那么相比于标准的卷积操作[13]，深度可分离卷积需要计算的参数数量要少很多，计算公式如下所示

$$\frac{H*W*C+C*N}{H*W*C*N}=\frac{1}{N}+\frac{1}{H*W} \tag{1}$$

其中，H 为图片的高度，W 为图片的宽度，C 为通道数，N 为通道数的个数.

本文在特征提取部分使用深度可分离卷积并非强行用精度换时间，在深度卷积的部分没有让通道之间的数据互相影响，逐点卷积又让通道间信息产生交互，用更低通道的特征图来存储特征信息. 在实际的实验中，谷歌团队的 MobileNet 相比于 VGG16 和 GoogleNet 在保持精度的情况下，大大减少了模型的计算参数，提高了效率. 综上所述，本文使用深度可分离卷积来进行行为特征识别工作.

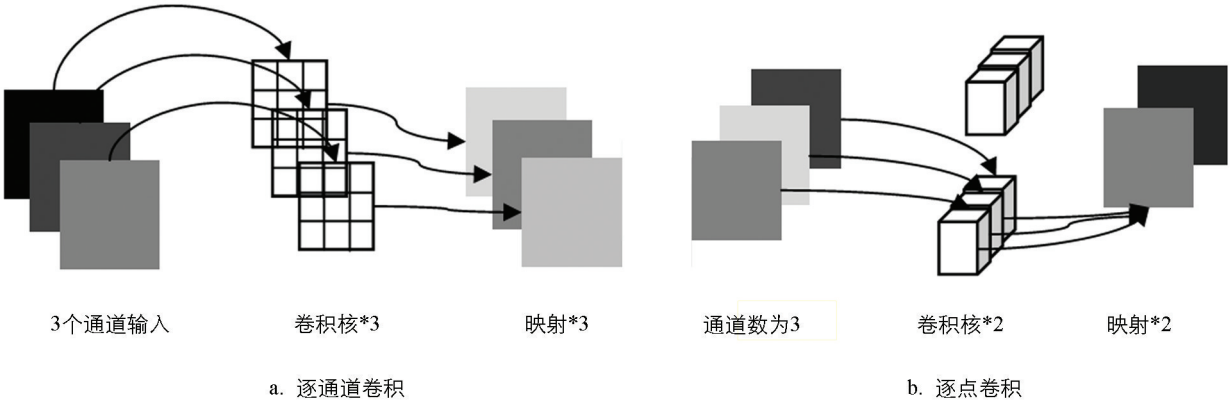

图 3　深度可分离卷积示意图

2.2　双域注意力

应用行为转化的灰度图其实也包含较多无关信息，例如在生成图片时用于补足长度的黑色部分，或者是流量信息的固定格式等. 为了降低无关信息的干扰影响，并且模型能在分类中的效果和效率都有所提升. 本文在卷积操作后引入了双域注意力模块，这样能在学习的过程中更加关注对于分类有帮助的信息. 本文使用的双域注意力模块如图 4 所示.

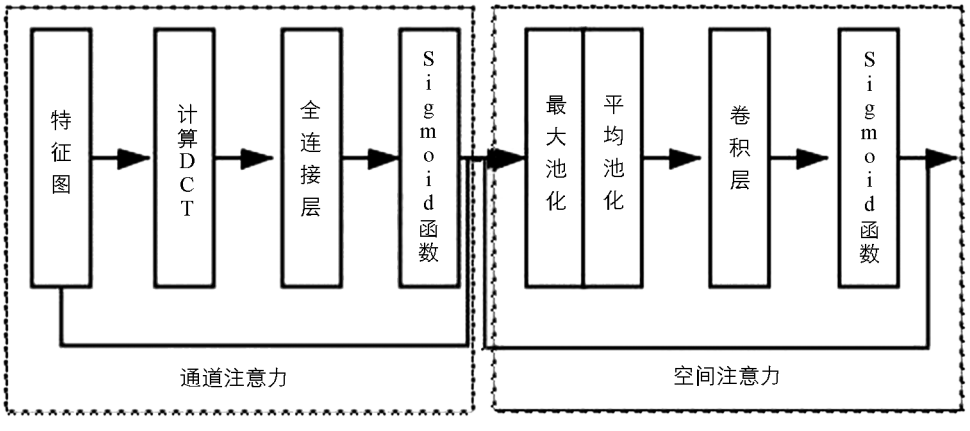

图 4　双域注意力模块

对于空间注意力来说，由于将每个通道中的特征都做同等处理，忽略了通道间的信息交互；而通道注意力则是将一个通道内的信息直接进行全局处理，容易忽略空间内的信息交互. 因此，将两者结合作用于本文的行为识别任务中.

1）通道注意力的计算

常规双域注意力中通道注意力的计算，首先是对输入特征图进行基于全局平均池化的预处理，然后进

行通道注意力的提取. 但是，从频域的角度进行分析后发现，全局平均池化只是频域中特征分解的特例，因此直接做全局平均池化会损失很多重要信息. 本文的通道注意力是基于DCT频域分析进行的，其计算公式为

$$f_{h,w}^{2d}=\sum_{i=0}^{H-1}\sum_{j=0}^{W-1}x_{i,j}^{2d}\cos\left(\frac{\pi h}{H}(i+0.5)\right)\cos\left(\frac{\pi w}{W}(j+0.5)\right)$$
$$h\in\{0,1,\cdots,H-1\},\ w\in\{0,1,\cdots,W-1\} \tag{2}$$

其中，$f_{h,w}^{2d}$ 表示二维DCT的频谱，h 和 w 分别表示输入特征图分量的高度和宽度，$x_{i,j}^{2d}$ 表示输入. 将特征图按照通道进行切分，则该特征图的频域注意力的预处理结果 $Freq^i$ 可以表示为

$$Freq^i=2DDCT(X^i)\qquad i\in\{0,1,\cdots,n-1\} \tag{3}$$

其中，X^i 表示第 i 个通道的特征图，n 表示总通道数，$2D$ 表示计算的是二维频域，DCT 表示将 X^i 作为输入，则 n 个通道的注意力预处理结果进行拼接后可以表示为

$$Freq=cat([Freq^0,Freq^1,\cdots,Freq^{n-1}]) \tag{4}$$

其中，cat 函数表示将多个元素收尾拼接起来，则通道注意力的计算公式为

$$Map_c(F)=\sigma(W_0(Freq)) \tag{5}$$

其中，$Map_c(F)$表示待生成的通道注意力，σ 表示Sigmoid激活函数，用来将注意力权重缩放为[0，1]，W_0 表示通道注意力中神经网络全连接层的参数，W_0 是需要进行学习的.

2) 空间注意力的计算

空间注意力将通道注意力模块的输出作为输入. 首先，将输入的特征图进行全局最大池化和全局平均池化处理，然后，将得到的结果进行拼接，经过一个卷积操作后降维到一个通道宽度的特征图，经过Sigmoid函数激活后生成空间注意力特征图，再与输入的特征图相乘，得到最后生成的特征图. 计算公式如下所示

$$Map_s(F)=\sigma(F^{7*7}([\mathrm{Avg}Pool(F);\mathrm{Max}Pool(F)])) \tag{6}$$

2.3 元学习

本文使用元学习[14-16]作为模型的学习器，即将含双域注意力的识别模型作为基础模型，再使用元学习的方式同时进行多个训练任务，然后通过获取不同任务合成的梯度方向来更新学习器.

一般的深度学习是将训练数据分为不同的批次，而对于元学习而言，是将训练数据分为不同的任务[17-19]，用于训练的任务称为训练任务 $T=\{T_1,T_2,T_3,\cdots\}$，而在每一个任务里又分了支持集和请求集，支持集对应了传统深度学习中的训练数据集，请求集则对应了传统深度学习中的测试数据集，在每一轮训练中，对于当前task，使用支持集来训练该模型，再用请求集去验证计算误差来更新学习器的参数. 用于测试的任务我们称为测试任务，那么对于这一部分的任务而言，也是分了2个数据集，即支持集和请求集，一个用于在 F 上训练找到一个适合这个分类任务的 f，另一个用于在实际的分类器上进行分类，检验产生的 f 的分类效果，从而评价 F 的学习能力[20]. 对于分类任务我们使用的损失函数是交叉熵损失函数，则

$$l=\frac{1}{N}\sum_i l_i=\frac{1}{N}\sum_i y_{ic}\lg(P_{ic}) \tag{7}$$

其中，N 指的是类别的数量，y 用0或者1来表示与该类是否相同，P 表示在识别模型中预测这个样本属于某个类别的概率.

在具体实现一轮训练任务的时候，首轮任务需要我们将学习器的参数初始化，并且需要提前设定好一个批次的任务数量，然后将这一批次的任务投入到识别网络中进行训练，用请求集的数据去计算在当前参

数 θ 下的损失，第二个任务采用同样的初始参数，重复上述的步骤再进行一次损失的计算，直到这一批次的任务训练结束，将计算的损失和作为我们学习器的损失，则

$$Loss(\varphi)=\sum_{n=1}^{N} l^{n}(\theta^{n}) \tag{8}$$

3　实验结果与分析

本文提出了一种基于双域注意力和元学习的移动应用行为识别方式，实现了在小样本情况下对移动应用行为的识别，使用了深度可分离卷积进行特征提取，并用双注意力机制对有用信息进行增益，另外还采用元学习的训练方式解决冷启动问题. 本文在实验中采用了自己采集的数据集，也采用了网络公开的数据集进行对比实验. 经过验证，本文提出的方法能够在小样本情况下进行移动应用行为的识别，并且在识别准确率上超过了其他小样本识别方法.

3.1　实验数据集

本实验的数据集是微信使用时所产生的网络数据流量. 流量的采集主要是通过自动化测试工具 Appium 来实现的，该工具能通过编写脚本自动执行相关操作. 使用 Appium 的原因是它的功能很强大且可以跨平台使用，这使得我们后期在采集安卓和 ios 的流量时可忽略系统的差异性.

在数据集构造方面主要分为 4 个步骤：

步骤一：流量采集. 流量采集使用的是基于 Appium 的自动化平台，通过自行编写的脚本在手机上执行既定的应用行为. 之后使用 Wireshark 进行抓包，获取 pacp 格式的文件. pacp 是一种常用的存储网络数据流的存储格式，大部分的抓包软件都可以获取这一类的数据. 对于元学习任务，本文采集的数据为微信的应用数据，共抓取了 6 种行为的流量，采集的用户行为种类如表 1 所示. 在实际使用中，每次使用 4 种行为作支持集，2 种行为作请求集，且假设测试用的样本为小样本.

表 1　数据集包含的行为种类

用户行为	数量	用户行为	数量
发文字	5 300	发图片	5 100
发红包	3 978	转账	3 978
点赞	5 499	发朋友圈	5 100

步骤二：流量筛选. 在进行流量采集的时候，尽管在自动执行的测试机上只运行了我们需要的移动应用，但是 Wireshark 在进行流量捕获的时候，会将所有流经网卡的网络数据流都保存下来，因此，得到的初始数据存在很多与实验无关的网络流量，此外，超时重传的数据包，在传输过程中丢失或者损坏的数据包，在连接初始用于 3 次握手没有携带数据的确认数据包，在数据传输阶段接收方发送的 ACK 数据包，以及用于断开连接的数据包，这些与实验无关的数据包都要进行过滤.

步骤三：流量切分. 依据 Taylor 等人[21]的论文观点，流量切分时依照不同的突发(Brust)阈值对流量进行切分，并依据目的 IP 和端口号进行流分离. 在实验中，我们采用相同的阈值来切分连续的行为流量，在后续也对时间阈值的选择进行了实验.

步骤四：生成流量图片. 取每个数据报的前 N 个字节，$N=1\ 024$，对于每一个图片而言，每个字节都对应了一个灰度，从而形成一张 $32*32$ 尺寸大小的灰度图片.

通过分析发现，不同的应用行为网络流量的差异性较大，例如，发送文字的数据包大小比发送图片的数据包小很多，而微信发红包、转账等行为的数据包集中在 300～1 300 B 左右，需要进一步分析识别.

3.2　实验任务抽样

虽然元学习解决的是小样本问题，但是在训练时仍需要大量的训练任务. 在传统的机器学习中，我们

用一个训练集去训练模型，再用另一个测试集去评判模型的质量，而对于元学习而言，我们不再专注于一个特定的任务，而是通过从数据集中抽样出一个任务集 $T=\{T_1, T_2, T_3, \cdots, T_n\}$，每次选取一个任务进行学习，而每个任务中都包含了训练样本(支持集)和测试样本(请求集)，通过对任务的训练，来实现让模型学习到一定的先验知识的目的，从而能够在新的识别任务中有好的识别效果.

在训练数据中，有 m 种不同类型的样本，从中随机选 $4(4<m)$个种类，每个种类 K 个样本，其中 $4K$ 个样本组成支持集(Support Set)，$2K$ 个样本组成请求集(Query Set). 在进行测试任务抽样时也采用相同的方式. 元学习的抽样算法如算法 1 所示.

算法 1 从数据集中生成小样本识别任务

输入：标签 $L=\{0, 1, 2, \cdots, N\}$

数据集 $D=\{(x_1, y_1), (x_2, y_2), \cdots, (x_n, y_n)\}$，$x_i \in R^d$，$y_i \in L$

$D\langle t\rangle$表示 D 中满足 $y_i=t$ 的所有元素(x_i, y_i)，$t \in L$

Query Set 大小为 B，每类样本数量为 K

输出：小样本识别任务 $T=\{Sa, Qu, K\}$

依赖：RandSample(P, R)表示从集合 P 中均匀且随机抽取 R 个样本

$La(la_0, \cdots, la_3)=$RandSample(L, 4)，$Lb(lb_0, lb_1)=$RandSample(L, 2)

生成支持集(Support Set)

for $i=1$ to 4：

$Sa\langle i\rangle=$RandSample ($D\langle la_i\rangle$, K)

$Sa=Sa \cup Sa\langle I\rangle$

end

生成请求集(Query Set)

$Qu\langle 0\rangle=$RandSample ($D\langle lb_0\rangle$, $B/2$)

$Qu\langle 1\rangle=$RandSample ($D\langle lb_1\rangle$, $B/2$)

$Qu=Qu\langle 1\rangle \cup Qu\langle 2\rangle$

$T=\{Sa, Qu, K\}$

3.3 评估指标

本文采用准确率(Accuracy)和召回率(Recall)作为评估模型的指标. 准确率用来评价模型对于行为识别整体的效果，如表 2 所示，其中 TP(True Positive)表示正例预测为正例的样本数；FP(False Positive)表示反例预测为正例的样本数；TN(True Negative)表示正例预测为反例的样本数；FN(False Negative)表示反例预测为反例的样本数.

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

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

其中，ACC 表示准确率，REC 表示召回率.

表 2 评估指标参数介绍

	正例	反例
识别为正	TP	FP
识别为反	TN	FN

3.4　模型收敛性

由于元学习的训练方式同一般的深度学习方式略有差别，因此首先需要确定本文提出的识别模型的收敛性，如图 5 所示展示了该行为识别模型在本文自行采集的数据集上的损失函数随迭代次数的变化.

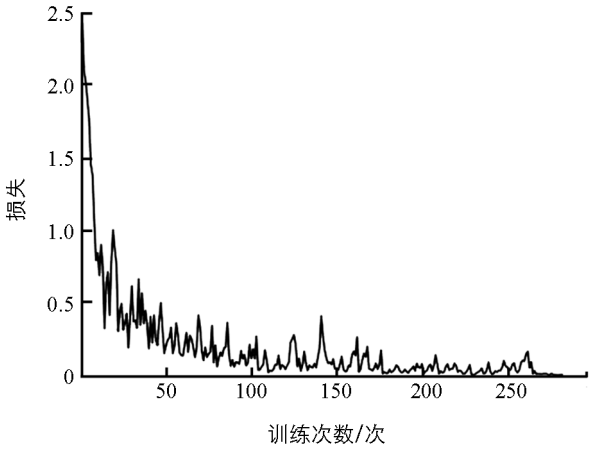

图 5　模型训练收敛

由图可知，在 100 次迭代之后模型趋于稳定，证明该识别模型是可以进行下一步实验的，并且我们将 100 作为后续实验的迭代次数.

3.5　时间阈值和样本数量影响

为了获得比较好的实验效果，本文对元学习样本参数的选取以及在流量的行为划分时时间阈值的选取进行了测试.

由图 6 可知，在进行流量的行为划分时，由于采取了不同的阈值，行为的划分直接影响了识别的准确率，同时，观察得知在时间阈值为 1.25 s 时可以取得最好的识别效果. 当小于 1.25 s 时存在将一个行为拆分开，而当大于 1.25 s 时存在将多个不同用户行为合并的情况，因此后续选取 1.25 s 作为时间阈值进行实验.

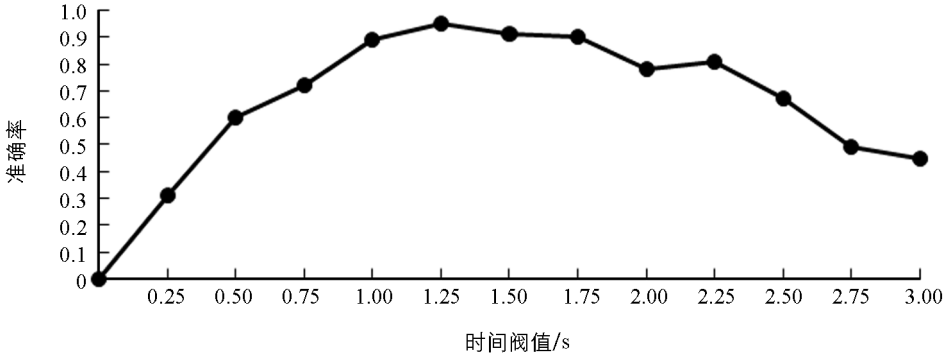

图 6　行为划分参数实验

另一个方面就是考虑在小样本的情况下，每次训练任务的样本数量对实验效果的影响，本文选取了 $K=5,10,15$ 这 3 种情况进行实验，都是小样本的场景，但是样本数量差别较大，实验结果如表 3 所示.

表 3　样本数量对实验结果的影响

类　型	$K=5$		$K=10$		$K=15$	
	ACC/%	*REC*/%	*ACC*/%	*REC*/%	*ACC*/%	*REC*/%
发文字	99.67	98.90	99.08	99.04	98.17	98.59
发图片	99.28	98.23	95.62	98.15	99.52	99.87
点赞	99.32	99.66	92.98	92.06	99.55	98.46
转账	98.17	97.78	98.73	98.43	96.38	96.13
发红包	98.01	98.44	98.49	97.98	97.36	98.02
发朋友圈	97.90	97.02	98.5	98.88	98.17	98.01

实验结果表明，在样本数量为 5 或者 10 时效果都比较好，模型具有泛化能力，由此可见元学习的学习方式是学习如何提取特征和学习如何进行比较，而不是记住了训练样本的特征，从而能在新的识别任务中有较好的表现.

3.6 注意力机制分析

本文采用的注意力机制是基于通道和空间的双域注意力机制，为了探究是否单独的通道或空间注意力会带来更好的效果，实验时采用了5种不同的注意力模块，如图7所示. 其中还引入了一个高效的通道注意力(ECA)模块来进行对比实验(如图7e)，该模块是一个轻量级注意力模块.

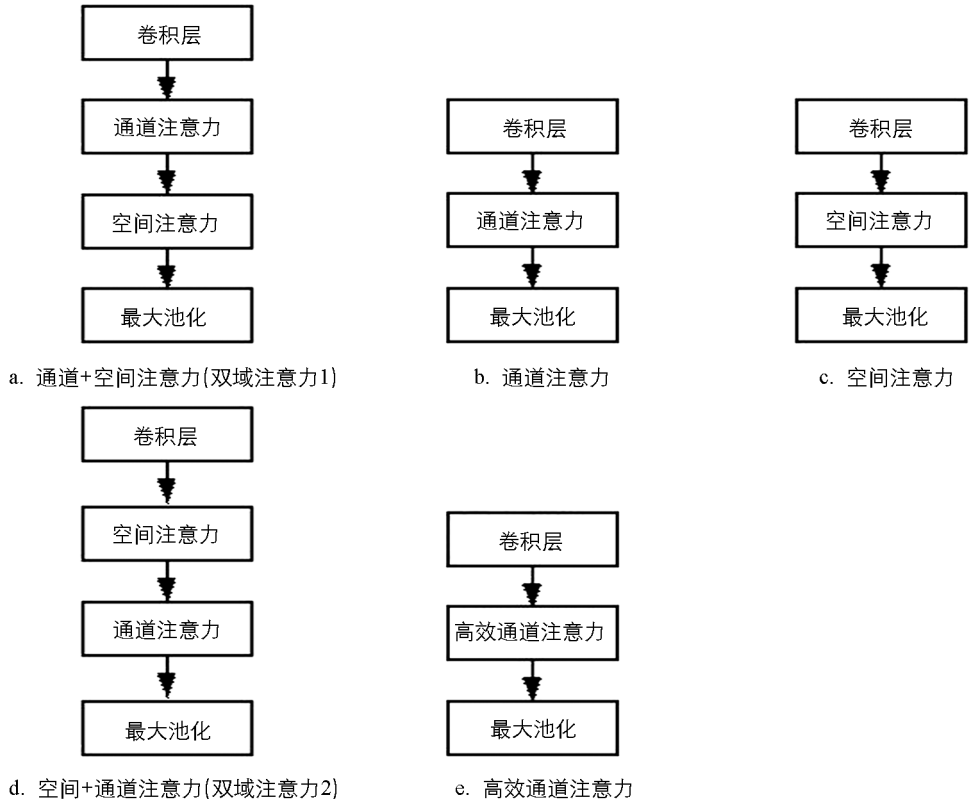

图7 不同注意力模块

将各类应用行为都按照每次任务10个样本，仍以4种行为举例，随机抽取训练任务和测试任务，最后得到的实验结果如图8所示.

实验结果表明不同的注意力模块确实会对实验结果产生影响，单一种类的注意力机制可能会忽视某些特征，而对于本文的行为识别，空间注意力机制对识别的效果影响要更大一些，对于类似长度的网络流量在形成灰度图后，观察图形的纹理会更有利于进行识别，而顺序的改变对实验结果基本没有影响，因此本文选取双域注意力机制构建识别模型.

3.7 泛化性分析

本文基于双域注意力机制和元学习的移动应用行为识别模式采用了元学习多任务的方式，因此模型在行为识别方面应具有一定的泛化性. 为了验证这一点，本文除了在自己所采集的数据集之外，还选用了网络公开的Instagram行为数据集，该数据集包括了4种行为：发文字消息、发图片、发布动态、点赞，与本文自行采集的微信行为有较大的相似性，因此在训练时采用微信的数据对模型进行训练，在测试的时候直接使用Instagram的行为数据集进行测试，实验结果如图9所示. 结果表明，本文提出的模型能够对Instagram的应用行为有较好的识别效果，表明了对于相似类型的移动应用，本文所提模型可以将学到的特征应用于新的小样本行为识别任务中.

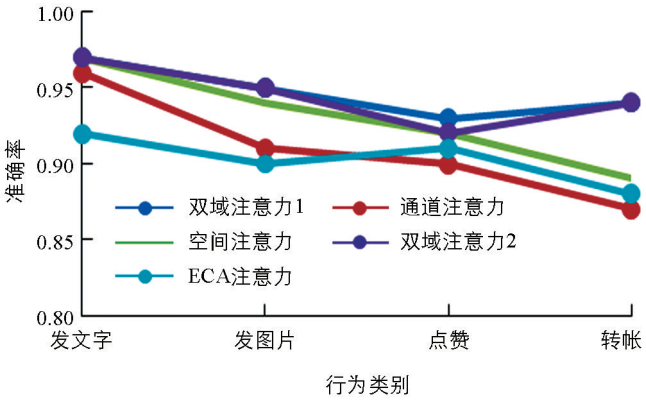

图 8　不同注意力模块实验结果

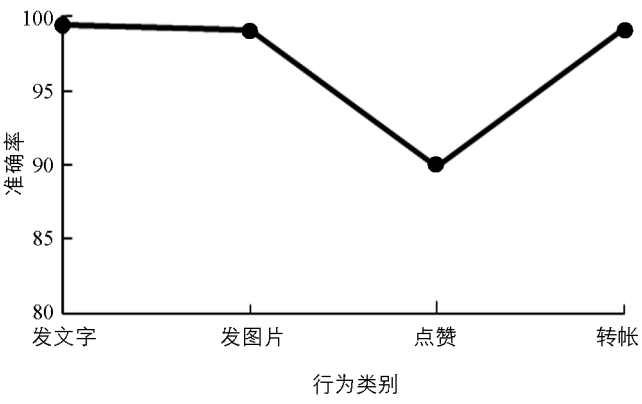

图 9　识别模型泛化性实验

3.8　实验结果

在小样本领域实现对移动应用行为的识别是一个比较新的研究领域，本文选取了用于行为识别的 CUMMA 模型以及专门用于小样本分类的 MAML 模型，进行对比实验，实验结果如图 10 所示.

可以看出，本文提出的小样本行为识别方法识别效果较好，对比实验中其他的模型和现有的行为识别模型，都需要大量样本进行训练，并且只针对特定类型的行为进行识别，可知本文的识别模型更具有实用性.

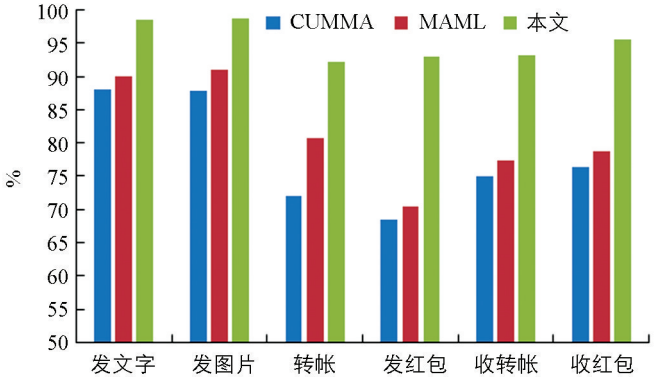

图 10　实验结果对比

4　结束语

移动智能设备和移动应用承载了诸多个人信息和日常工作娱乐功能，通过分析移动应用在使用时产生的网络流量可以在网络管理、隐私保护以及行为识别方面提供有价值的信息. 本文针对小样本场景下的移动应用行为识别方式进行了研究，提出了一种基于双域注意力机制的行为识别方式，该方式提高了移动应用行为识别的准确性；文章还采用了元学习的训练方式，解决了小样本场景下，移动应用行为识别冷启动的问题. 当然，本文提出的应用行为识别方式还存在不足和需要改进的地方，使用深度可分离卷积虽然降低了计算量，但注意力机制会增加一部分计算量，因此还需要进一步深入研究计算资源消耗的问题，寻找更加高效的注意力实现方式.

参考文献：

[1] GUO YY, WANG W P, ZHANG H, et al. Traffic Engineering in Hybrid Software Defined Network via Reinforcement Learning [J]. Journal of Network and Computer Applications, 2021, 189: 103116.

[2] 赵　颖，王　权，黄叶子，等. 多视图合作的网络流量时序数据可视分析 [J]. 软件学报，2016，27(5)：1188-1198.

[3] ALAN H F, KAUR J. Can Android Applications be Identified Using only TCP/IP Headers of Their Launch Time Traffic? [C] //Proceedings of the 9th ACM Conference on Security & Privacy in Wireless and Mobile Networks. New York, NY, USA: ACM, 2016: 61-66.

[4] WANG Z J, DONG Y N, MAO S W, et al. Internet Multimedia Traffic Classification from QoS Perspective Using Semi-Supervised Dictionary Learning Models [J]. China Communications, 2017, 14(10): 202-218.

[5] SHAFIQ M, YU X Z, BASHIR A K, et al. A Machine Learning Approach for Feature Selection Traffic Classification

Using Security Analysis [J]. The Journal of Supercomputing, 2018, 74(10): 4867-4892.

[6] COULL S E, DYER K P. Traffic Analysis of Encrypted Messaging Services: Apple Imessage and Beyond [J]. ACM SIGCOMM Computer Communication Review, 2014, 44(5): 5-11.

[7] LEE K M, PARK K S, HWANG K S, et al. Deep Neural Network Model Construction with Interactive Code Reuse and Automatic Code Transformation [J]. Concurrency and Computation: Practice and Experience, 2020, 32(18): 1002.

[8] LI D, LI W Z, WANG X L, et al. App Trajectory Recognition over Encrypted Internet Traffic Based on Deep Neural Network [J]. Computer Networks, 2020, 179: 107372.

[9] GU C J, ZHANG S Y, XUE X Z. Encrypted Internet Traffic Classification Method Based on Host Behavior [J]. International Journal of Digital Content Technology and Its Applications, 2011, 5(3): 167-174.

[10] NAN Y H, YANG Z M, YANG M, et al. Identifying User-Input Privacy in Mobile Applications at a Large Scale [J]. IEEE Transactions on Information Forensics and Security, 2017, 12(3): 647-661.

[11] 王 伟. 基于深度学习的网络流量分类及异常检测方法研究 [D]. 合肥: 中国科学技术大学, 2018: 69-90.

[12] XU Z X, CHEN X L, TANG W, et al. Meta Weight Learning via Model-Agnostic Meta-Learning [J]. Neurocomputing, 2021, 432: 124-132.

[13] CONTI M, MANCINI L V, SPOLAOR R, et al. Analyzing Android Encrypted Network Traffic to Identify User Actions [J]. IEEE Transactions on Information Forensics and Security, 2016, 11(1): 114-125.

[14] ESPINAL A, ESTRADA R, MONSALVE C. Traffic Model Using a Novel Sniffer that Ensures the User Data Privacy [J]. MATEC Web of Conferences, 2019, 292: 03002.

[15] LIU X Q, ZHOU F Y, LIU J, et al. Meta-Learning Based Prototype-Relation Network for Few-Shot Classification [J]. Neurocomputing, 2020, 383(6): 224-234.

[16] LANDRO N, GALLO I, GRASSA R L. Combining Optimization Methods Using an Adaptive Meta Optimizer [J]. Algorithms, 2021, 14(6): 186.

[17] MUNKHDALAI T, YU H. Meta Networks [J]. Proceedings of Machine Learning Research, 2017, 70: 2254-2257.

[18] QIAN Q, JIN R, YI J F, et al. Efficient Distance Metric Learning by Adaptive Sampling and Mini-Batch Stochastic Gradient Descent (SGD) [J]. Machine Learning, 2015, 99(3): 353-372.

[19] LI H, YANG X, LI Y, et al. Evolutionary Extreme Learning Machine with Sparse Cost Matrix for Imbalanced Learning [J]. ISA Transactions, 2020, 100: 198-209.

[20] XU J C, DU Q F. Learning Transferable Features in Meta-Learning for Few-Shot Text Classification [J]. Pattern Recognition Letters, 2020, 135: 271-278.

[21] TAYLOR V F, SPOLAOR R, CONTI M, et al. Robust Smartphone App Identification via Encrypted Network Traffic Analysis [J]. IEEE Transactions on Information Forensics and Security, 2017, 13(1): 63-78.

责任编辑 崔玉洁