基于对比学习的加密流量编码器

1引言

深度学习的一大优势就是可以自动从输入数据中提取特征并进行学习。虽然 这种方法可以为研究者节约特征工程的时间，但是目前已知的很多深度学习模型 提取特征时并不会对特征进行明确的过滤或筛选，导致一些低价值的特征被保留 了下来。这里的低价值特征，主要指的是难以反映数据的本质的特征。这些特征要么是很多类别的数据都具有，要么是某几个数据特有的。这两种特征都无法很好 地代表一类数据。因此，当这些特征被保留下来后，模型针对某个任务的学习有可能会被误导。文献[4] 中提供了一个自动驾驶领域的例子：对于同一个模型，在车前有行人的情况下判断是否应该踩下刹车时，是否提供控制台的信息会导致两种截然不同的结果。而错误的结果，即车前有行人却不踩刹车，是在为模型提供控制台信息后得到的。这里的控制台信息就是一种难以反映当前情况本质的低价值特征。该特征不但多余，而且还干扰了模型的判断。

当使用深度学习模型进行加密流量分类时，尽管很多方法都被证明是行之有效的，但是目前也没有相关的研究关注模型对特征提取的控制。不过导致这种现象的原因更多还是在于模型本身的训练机制。对于有监督模型而言，如图1所示，特征在被神经网络提取出来后就会直接参与分类。神经网络会通过计算将提取到的特征与一个人工设置的数字标签进行拟合。

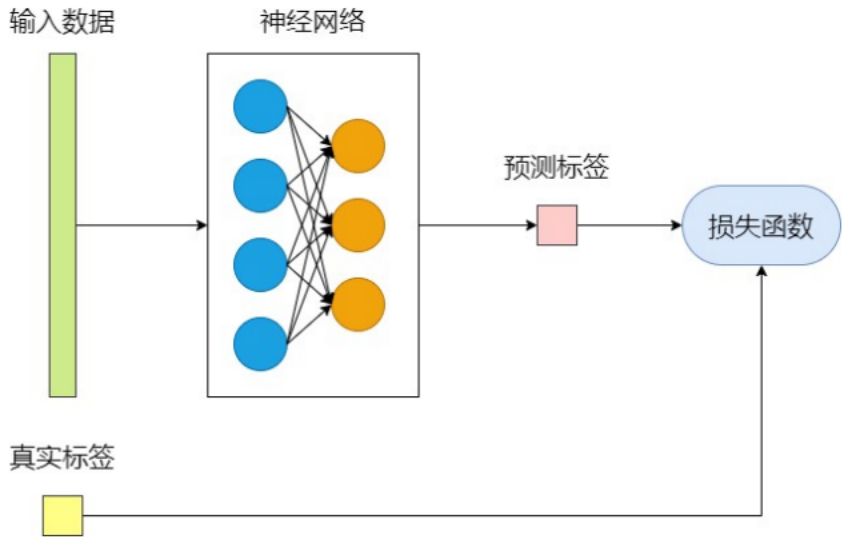


图 1　端到端的有监督学习过程

它的整个过程是端到端的，并没有表征向量可以提取，也难以被人工干预。所以让模型关注特征筛选是办不到的。另外，这种训练方式更多是一种面向拟合的特征提取。虽然提取到的特征可能有利于分类，但是这些特征可能只是恰好有利于拟合某类标签而与数据的本质关系不大。对于半监督模型而言，虽然上游的自监督学习任务可以帮助提取表征向量，但是目前的半监督学习多是基于AE[44-45,52]来提取表征向量的，如图2所示。

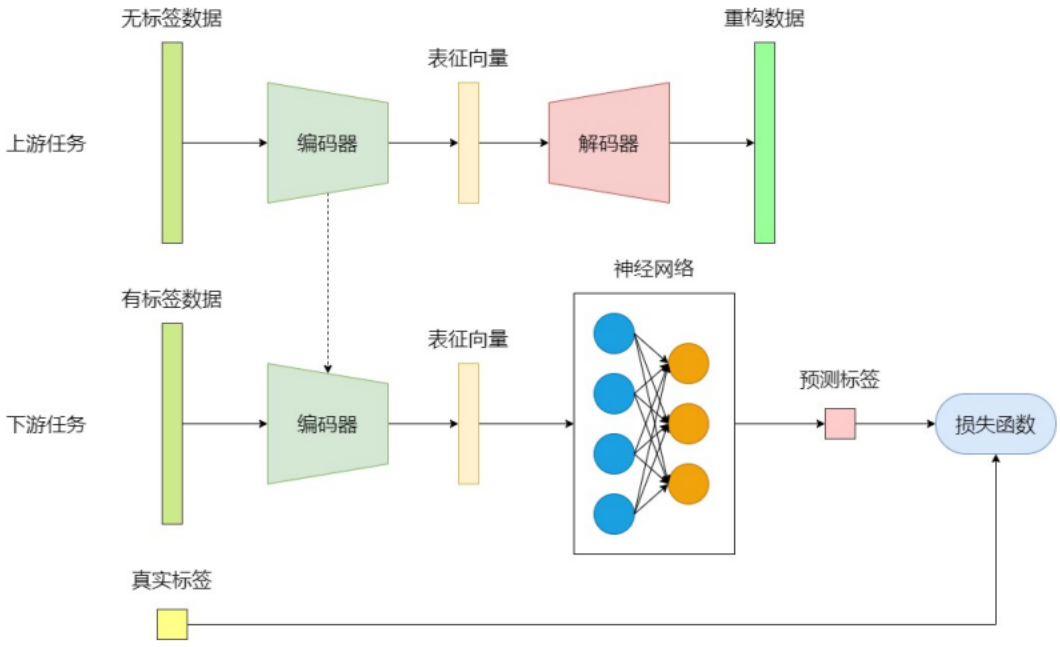


图 2　基于 AE 的半监督学习过程

AE的编码器在提取表征向量时为了尽可能地缩小输入数据和重构数据之间的差异，会尽可能多地提取数据特征。这些特征虽然有利于了解数据的全貌，但是并非都是必需的。另外，AE在提取特征时只会一对一地比较输入数据与重构数据，不会将输入数据与其他数据进行比较。实际上，横向地对比其他相似或不相似的数据也有利于模型发掘关键特征和过滤掉低价值特征。

而同作为半监督学习的CL不但可以提取表征向量还可以在提取的过程中保证对特征进行过滤和筛选，使得表征向量中留下的多是能够反映数据本质的关键特征。本文的贡献有以下几点：

1. 提出了基于CL的加密流量数据编码器以及该编码器的训练框架。在训练阶段，编码器通过对同训练批次的增强样本进行比较，可以过滤掉低价值特征。为从加密流量数据中提取更加高质量的表征向量提供了思路。
2. 针对加密流量数据自身的特性，提出合适的数据增强方法。

3. 采用可视化技术和计算相关量化指标的方式初步证明相对于基线模型，基于CL的编码器的确可以对特征进行过滤和筛选并且可以提升表征向量的质量。

2 训练框架

整个编码器的训练框架如图3所示。我们记数据预处理后，由同一个训练批次的输入数据组成的集合为B。对于任意一个B中的输入数据xi，框架会先将 xi 进行两次复制，得到多个复制数据。增强器会对多个数据分布进行处理得到与输入数据相似但不完全相同的增强样本。编码器会从增强样本中提取特征并将其压缩成表征向量。投射器对表征向量进行进一步地过滤压缩得到嵌入向量。损失函数会以所有嵌入向量作为输入计算嵌入向量各自的损失函数值。B中所有输入数据的增强样本的损失函数值的算数平均数即该训练批次的损失函数值$Loss$。通过最小化$Loss$可以实现对分类器的训练。

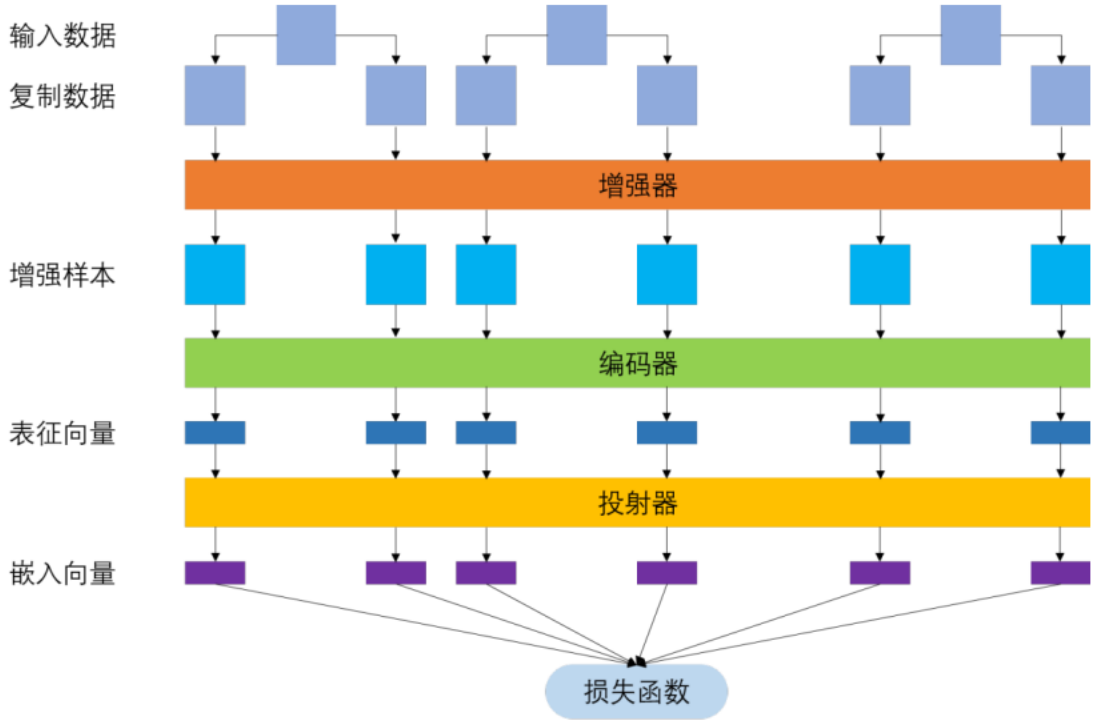


图 3　基于CL 的加密流量编码器的训练框架

本文在下面将介绍训练框架中各个模块的设计细节。

3 数据预处理

无论是机器学习模型还是深度学习模型，数据的输入都需要遵守一定的规范。这种规范是由选取的模型和设定的超参数来决定的。本文不考虑使用RNN模型作为编码器。一方面是因为在真实网络环境中收集完整会话或者数据流会严重增加网络设备的负担，是很不现实的。另一方面是因为以字节序列作为RNN的输入粒度太小。因此，本文倾向于使用CNN模型作为编码器。而 CNN 模型的输入要求各个维度大小是固定的。因此，参考相关工作[26-27, 36, 38]的做法，本文也对数据流的前 784 个字节进行提取。这样做有以下几个好处：第一，数据流的前 784 个字节包含第一个数据包的大部分甚至全部内容，其中包含头部的明文信息。从这些明文信息中，编码器可以学习关于该流量数据更加具体明确的内容。第二，长度为 784 的字节流可以很容易地映射成 28*28 的图像。这种尺寸的二维图像可以直接被很多经典的CNN模型处理。第三，排除数据维度不同的因素更利于对比不同模型或者训练方法之间的差异。

本文的数据预处理过程如图4所示。首先，对以类别标签命名的 pcap 流量文件进行切分，切分粒度为数据流（可以认为一个完整的会话是由多个不同的数据流组成的）然后，提取数据流的前 784 个字节。其中，超出 784 字节的部分直接丢弃，不足的剩余部分用ASCII码0来补全。接着，将每个字节按照对应的 ASCII码值映射成[0, 255]中的整数。然后，按照数值对应的灰度值将整数序列转化为单通道的灰度图像。最后，将灰度图像保存备用。

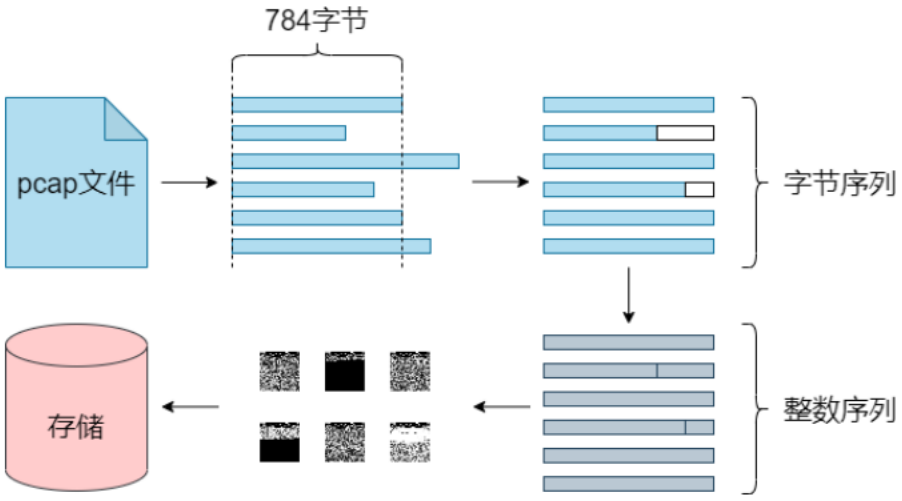


图 4 数据预处理过程

4 训练组件

4.1 增强器

增强器的作用是对数据进行数据增强操作，得到多个与输入数据相似但不完全相同的增强样本。实际上，数据增强在其他机器学习或深度学习的任务中的主要作用是扩充次要类别数据的规模使模型可以在一个各类数据分布相对均匀的情况下学习。但是在本文中，数据增强操作是必不可少的工作。因为关键特征通常对数据增强操作具有一定的不变性[51] 。

在CV 领域，数据增强操作十分常见，甚至已经集成到了很多成熟的深度学习框架中。所以对于CV领域的数据做数据增强是没有必要额外设计增强器的。但 是，本文编码器使用的输入数据来自于字节序列。这就意味着输入数据自身是具 有一种语义的。尽管被转化成了灰度图像，但是这种语义并没有随之消失。因此，如果依旧采用CV 领域的数据增强方法包括旋转、拉伸、裁剪和改变色调等操作很可能会完全破坏掉这种语义。因此，在设计增强器或者探究适用于加密流量数据 的增强方法时，需要在保证能获得多个增强样本的同时对数据的语义做小程度的 破坏。

既然无法完全借鉴CV领域的数据增强方法而且数据自身又具有语义，那么自然会转而考虑借鉴NLP 领域的数据增强方法。在 NLP 领域，比较基础的数据增强方法包括同义词替换、随机插入、随机交换和随机删除[53] 。实验也证明，虽然以上数据增强方法会对语义造成一定的破坏，使得句子不易理解，但是模型的鲁棒性反而得到了增强，而且增强后的数据与原数据在特征空间上仍然很接近。

然而，联系到本文的应用场景，对于加密流量数据而言，不存在所谓的同义词，所以不能采用同义词替换。随机交换会破坏数据不止一处的位置。随机插入和随机删除会改变数据长度，因此也不能直接使用。不过，可以将随机插入和随机删除结合在一起使用。首先，选取某个位置，将从该位置开始往后固定长度的数据 删除。然后，还是在该位置插入ASCII 码0（也是十进制数字 0）直到数据恢复成了原来的长度。这种做法最终的效果等同于先在数据中选取固定长度的连续位置，然后将这些位置对应的数据用 0 覆盖掉。因此本文将这种数据增强方法叫做数据随机覆盖。

数据随机覆盖既保证了可以在获得多个增强样本的同时对数据语义做最小程度的破坏，又保证了数据格式依旧满足 CNN 的输入要求。

4.2 编码器

出于诸多因素的考虑CNN 模型比RNN 模型更适合作为编码器。更进一步地，由于一维 CNN 的卷积核和二维 CNN 的卷积核扫描数据的方式有很大的不同（如图

3.5所示），而一维 CNN 卷积核的扫描方式更加适用于像加密流量数据这样的序列类型数据[26] 。因此，本文将使用一维 CNN 模型实现编码器。整个编码器的结构参数如表 3.1所示。其中，ReLU[54]是一种非线性激活函数，其表达式如公式（1）所示。引入非线性激活函数的好处是可以令神经网络学习非线性关系。而 ReLU 函数与其他常见的激活函数（如Sigmoid 函数和Tanh函数）相比一方面计算量要更小，另一方面不容易产生梯度消失现象。

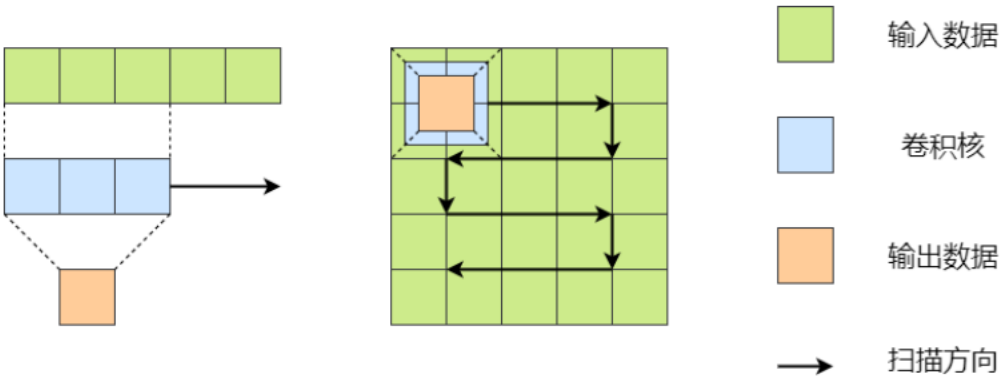


图 5 一维 CNN 模型（左）与二维 CNN 模型（右）卷积核的扫描方式

最大池化操作直观的理解就是将选定数据区域中最大的值提取出来。该操作可以保证在不丢失大量有用信息的情况下缩小特征图的大小从而实现数据降维，最后减少训练所需的参数量，提升训练效率。

另外，步长指的是卷积核每次沿着扫描方向移动的跨度。填充长度则是指在输入数据两端填充 0 的长度。在输入数据两端填充 0 的目的是为了尽量充分地获取数据边缘的信息。

表1 基于CL 的编码器的结构参数

操作	输入维度	卷积核大小	步长大小	填充长度	输出维度
一维卷积+ReLU	1*784	25	1	12	32*784
最大池化	32*784	3	3	-	32*262
一维卷积+ReLU	32*262	25	1	12	64*262
最大池化	64*262	3	3	-	64*88
全连接层+ReLU	5632	-	-	-	1024
全连接层	1024	-	-	-	588

编码器最终会输出一个长度为588的表征向量，该表征向量将在下游任务中发挥作用。但是在上游的辅助任务中，表征向量还需要输入到投射器中做进一步处理。

4.3 投射器

投射器从操作上来说就是将表征向量进行进一步地映射和压缩，得到一个维度更更低的嵌入向量。这种操作虽然看起来只是希望通过低维度的向量加速损失函 数的计算，但是它还有另外一个很重要的功能，那就是防止编码器为了过度迎合损失函数的优化而过滤掉关键特征。文献[48] 曾指出，投射器的存在可以让更多有价值的信息保留在表征向量中。该结构相当于对特征进行了二次过滤，让由数据增强产生的差异保留到嵌入向量中。而损失函数在指导编码器优化的过程中会尝试消除这种差异，从而保证让源自同一个输入数据的增强样本的表征向量更加接近。一般的对比学习模型通常会使用简单的 MLP 作为投射器[51] 。本文也同样采用这种结构作为投射器。投射器的具体结构参数如表2所示。

表2 投射器的结构参数

操作	输入维度	输出维度
全连接层+ReLU	588	392
全连接层	392	196

5 实验数据集

为了得到更加丰富的实验结果，同时也为了增强结果的说服力，本文尝试在 3个不同的数据集上进行实验。其中两个数据集是开源数据集 USTC-TFC2016[27] 中的两个子数据集，本文将其分别记为 TFB 和TFM。这两个子数据集各自有 10 个应用类别的流量数据。由于这两个子数据集的类别标签不存在重复现象且数据的收集方式也

不相同，因此可以将它们看作两个独立的数据集。还有一个包含 10 个应用类别的移动流量数据集是基于清华校园网的网络环境收集的，记作 THC。THC数据集的收集需要借助虚拟网卡和移动热点技术。整个过程需要运行特定应用的移动端连接指定的移动热点，然后再通过 Wireshark[56]软件监听虚拟网卡，从而得到包含特定应用的流量的数据文件。

实验使用的3个数据集的统计信息如表3所示，其中类别是具体的应用名，规模则是数据流的个数。

表 3　　实验数据集的统计信息

TFB		TFM		THC	
类别	规模	类别	规模	类别	规模
BitTorrent	15,000	Cridex	461,000	Airbnb	2,000
Facetime	6,000	Geodo	213,000	BiliBili	1,000
FTP	360,000	Htbot	169,000	Gaodemap	5,000
Gmail	25,000	Miuref	81,000	JD	6,000
MySQL	200,000	Neris	498,000	Kuaishou	2,000
Outlook	15,000	Nsis-ay	352,000	Netease	3,000
Skype	12,000	Shifu	500,000	QQ	2,000
SMB	925,000	Tinba	22,000	Taobao	3,000
Weibo	2,610,000	Virut	438,000	Weibo	2,000
World of Warcraft	140,000	Zeus	86,000	Zhihu	3,000

6 基线模型

由于 CL 本身是自监督学习中的一种，同时在学术界中另外一种常用来提取表征向量的自监督模型是AE[44-45]。因此，本文将以基于 AE 的加密流量编码器作为基线模型。模型训练框架的结构示意图参考图。其中，为了消除不同编码器结构对结果的影响，基于 AE 的编码器在结构参数上与基于 CL 的编码器完全相同。这里只给出解码器的结构，如表4所示。

对于神经网络的训练结果，除了网络结构外，超参数的选取也会对其产生重要的影响。增强器方面，数据随机覆盖的字节长度为 14。优化器方面，两个模型均采用 Adam[57]优化器。其中梯度指数平均衰减率 $\beta 1$ 取 0.5，梯度平方指数平均衰减率$\beta 2$ 取 0.99。学习率方面，经过多次尝试，除了基线模型在用 TFM 数据集训练时学习率取 0.0001，其他实验中学习率均取 0.001。训练批次大小方面，所有实验都保证为 100。最后，保

证两种模型的编码器都会训练 100 个迭代轮次。

表4 基于 AE 的解码器的结构参数

操作	输入维度	卷积核大小	步长大小	填充长度	输出维度
全连接层+ReLU	588	-	-	-	1024
全连接层	1024	-	-	-	5632
转置卷积+ReLU	64*88	3	3	1	32*262
转置卷积+Tanh	32*262	3	3	1	1*784

7 实验过程与结果分析

7.1 验证编码器能否对特征进行过滤和筛选

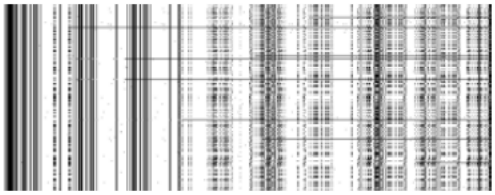
对于机器学习模型而言，由于输入数据中的每个元素或若干元素的组合都可以视为一种特征，因此评估特征的重要性并对特征进行过滤和筛选是一件相对容易的事情。例如，逻辑回归中可以根据各个特征分配的权重参数大小来判断，决策树模型中可以通过可视化节点分裂条件来分析，还有一些模型可以通过计算 Shapley[58] 值的方式来衡量各个特征对结果的贡献。

然而，在本文的场景中，由于模型的输入并非事先提取好的特征，或者说模型学习的特征是高度抽象的，因此上述的做法很难应用到本文的研究中。不过，庆幸的是，目前也存在一些可视化方法[59-60] 可以在直观的角度上展现深度学习模型对特征进行提取的过程。在这里，本文尝试可视化两种编码器的中间特征图。所谓中间特征图就是将网络中间某层的输出视作图片进行可视化展示的结果。

在本文实验中，我们分别提取了两种编码器中第一层神经网络的输出并将这些输出结果制成中间特征图。之所以选择第一层是因为该层与输入数据之间是直接接触的，而且该层提取的特征也是最贴近数据本身的。这种方法虽然无法帮助确定编码器究竟提取的是什么特征，但是可以揭示编码器在提取特征时更加看重哪些位置。

我们的实验过程如下：首先利用无标签数据训练编码器。然后从各个类别的数据中随机抽取 300 条数据作为编码器的输入。编码器在对同一类别的 300条数进行处理后，将每条数据经过第一层神经网络后得到的所有中间特征图保存下 来。由于数据在经过神经网络后相当于单通道图像变成多通道图像，而每一个通道又各自对应一个中间特征图，因此我们会将这些中间特征图进行叠加得到一个新的中间特征图。最后，在编码器处理完同一个类别的所有数据后，这些同类别数据对应的特征图会一并输出展示。

本文在此展示1种类别的数据的实验结果，如图6所示。该数据来自 TFB 数据集。每幅图中的每个子图都是 300 行的。输入数据可视化子图中每一行代表一条输入数据，而中间特征图子图中每一行代表同行输入数据对应的中间特征图。



(a) 输入数据可视化

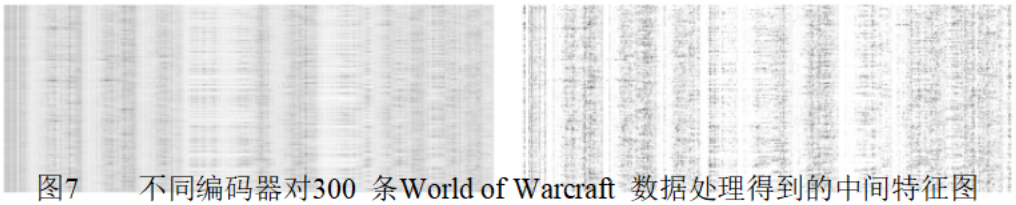


图7　　不同编码器对300 条World of Warcraft 数据处理得到的中间特征图

(b) 基于 AE 的编码器的中间特征图。(c) 基于 CL 的编码器的中间特征

对于中间特征图而言，某个部位颜色越深，意味着编码器对数据中对应的部位会给予更高程度的关注。换句话说，编码器会认为从数据的该部位可以提取出更有价值的特征。对比两种编码器得到的中间特征图，我们可以得到以下结论：

1. 两种编码器得到的中间特征图中深色区域存在大部分重叠。这说明二者都认为从这些区域中可以提取到可用的特征。

2. 基于 CL 的编码器得到的中间特征图中存在很多白色的区域，而这些区域在基于 AE 的编码器得到的中间特征图中多是浅灰色的。这说明基于 AE 的编码器在提取特征时会尽可能多地考虑到数据的每个部分，但这也意味着基于 AE 的编码器并不会对特征进行进一步筛选和过滤。而基于 CL 的编码器则会把注意力集中在更能提供有价值特征，即关键特征的位置上，但对于其他不那么重要的位置则近乎无视。产生这一差异的主要原因在于两种编码器的训练机制的不同。这种训练机制上的差异已在本文第 2.3节中进行过论述。

7.2量化分析表征向量质量

实际上，目前在学术界尚且没有十分完备的方法可以用来综合评定表征向量的质量。因此，我们尝试引入 3 种不同的指标，从不同的角度来比较两种编码器得到的表征向量。

前两种指标是 Wang 等人[61] 提出的 Alignment 和Uniformity。Wang 认为，假如要将对比学习模型得到的表征向量映射到一个单位超球体的球面上，那么正样本对的表征向量应该尽可能地接近。这种接近程度可以用 Alignment 来衡量。同时所有的表征向量在单位超球体球面上的分布应该更加均匀。而这种均匀程度可以用

Uniformity 来衡量。根据它们的含义，可以尝试利用这两个指标来衡量同类数据表征向量之间的接近程度以及不同类数据表征向量之间的疏离程度。二者都是取值越小说明程度越大。对于K-Means 算法而言，同类簇内部数据的协方差越小越好，不同类簇之间的协方差越大越好，即 CH 值越大越好。CH 取值越大即说明同类簇内的数据分布更加紧密， 而不同类簇之间更加分散。本文之所以将CH也作为一种参考指标也是考虑到该 指标的含义与对比学习的本质很接近。

我们的实验过程如下：首先利用无标签数据训练编码器。然后从各个类别的 数据中随机抽取 300 条数据作为编码器的输入。编码器在对所有数据进行处理后，将得到的所有表征向量全部保存起来。这些表征向量将有两个去处，一个是用来 计算 Alignment 和 Uniformity，另一个是作为 K-Means 模型（K 设置为 10）的输入。当 K-Means 模型训练完毕后我们会计算该模型的 CH 指数。

实验结果如表5所示。

表 5 表征向量质量量化分析结果

指数	Alignment		Uniformity		CH 指数	
	CL	AE	CL	AE	CL	AE
TFB	1.9432	1.1782	-3.4339	-2.2004	1776.8953	130.0426
TFM	1.8420	1.4953	-3.2405	-2.9600	212.4564	149.4520
THC	1.9367	1.4152	-3.5519	-2.7445	158.6324	101.0759

分析表5，我们可以得到以下结论：

1. 从Alignment 上看，基于 CL 的编码器提取的表征向量要不如基于 AE 的编码器提取的表征向量。这说明基于 AE 的编码器提取的表征向量中正样本对的表征向量更加接近。产生这种结果的原因在于基于 CL 的编码器在根据公式（3.12）进行优化时会将负样本对之间的表征向量推开。不过由于基于 CL 的编码器在训练时只会将源自同一个输入数据的增强样本对视作正样本对， 其余的都视作负样本对，因此在训练时也难免会将本是同一类的数据的表征向量推开。而基于 AE 的编码器在训练时不会故意将表征向量拉近或者推开。所以产生该结果是在情理之中的。
2. 从Uniformity 上看，基于 CL 的编码器提取的表征向量要优于基于 AE 的编码器提取的表征向量。根据这个指标的含义，我们可以认为基于 CL 的编码器可以使表征向量在单位超球体球面上的分布更加均匀。表征向量分布更加均匀意味着不同类别数据的表征向量差异性可能增大。当不同类别数据的表征向量差异性增大，意味着它们更容易被区分。从这种角度上看，基于 CL的

编码器确实对表征向量质量的提升做出了一定贡献。

3. 从 CH 指数上看，基于 CL 的编码器提取的表征向量明显要比基于 AE 的编码器提取的表征向量更能保证同类簇的数据更加紧密同时不同类簇的数据更加分散。

8小结

本文提出了一种基于CL 的加密流量编码器。该编码器以数据流的前 784 个字节作为输入，可以输出其对应的表征向量。为了得到包含更多关键特征和更少低价值特征的表征向量，基于 CL 的编码器会对比同一个训练批次中的所有输入数据的增强样本。本文采用可视化技术和计算相关量化指标的方法，与基于 AE 的编码器进行对比，初步验证了基于 CL 的编码器确实可以对特征进行过滤和筛选并在一定程度上提升表征向量的质量。

参考文献

[1] Google. Google 透明度报告[EB/OL]. 2022[2022-03-06]. https://transparencyreport.google.com/https/overview.

[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]. 信息网络安全,2019(3): 7.

[3] Anderson B, McGrew D. Identifying encrypted malware traffic with contextual flow data[C]// Proceedings of the 2016 ACM workshop on artificial intelligence and security. 2016: 35-46.

[4] De Haan P, Jayaraman D, Levine S. Causal confusion in imitation learning[J]. Advances in Neural Information Processing Systems, 2019, 32.

[5] Dainotti A, Pescape A, Claffy K C. Issues and future directions in traffic classification[J/OL]. IEEE Network, 2012, 26(1): 35-40. DOI: 10.1109/MNET.2012.6135854.

[6] Constantinou F, Mavrommatis P. Identifying known and unknown peer-to-peer traffic[C/OL]// Fifth IEEE International Symposium on Network Computing and Applications (NCA'06). 2006: 93-102. DOI: 10.1109/NCA.2006.34.

[7] Thay C, Visoottiviseth V, Mongkolluksamee S. P2p traffic classification for residential network [C/OL]//2015 International Computer Science and Engineering Conference (ICSEC). 2015: 1-6. DOI: 10.1109/ICSEC.2015.7401433.

[8] Hafeez S. Deep packet inspection[J]. Eetimes Com, 2016(11): 585-589.

[9] Haffner P. Acas : Automated construction of application signatures[C]//SIGCOMM '05 Workshops, Augst. 2005.

[10] Wright C, Monrose F, Masson G M. Hmm profiles for network traffic classification[C]//Acm Workshop on Visualization & Data Mining for Computer Security. 2004: 9.

[11] Wright C V, Monrose F, Masson G M. On inferring application protocol behaviors in encrypted network traffic[J]. Journal of Machine Learning Research, 2006, 6(4): 2745-2769.

[12] Alshammari R, Zincir-Heywood A N. Machine learning based encrypted traffic classification: Identifying ssh and skype[C/OL]//2009 IEEE Symposium on Computational Intelligence for Security and Defense Applications. 2009: 1-8. DOI: 10.1109/CISDA.2009.5356534.

[13] Alshammari R, Nur Zincir-Heywood A. A flow based approach for ssh traffic detection[C/OL]// 2007 IEEE International Conference on Systems, Man and Cybernetics. 2007: 296-301. DOI: 10.1109/ICSMC.2007.4414006.

[14] Dusi M, Este A, Gringoli F, et al. Using gmm and svm-based techniques for the classification of ssh-encrypted traffic[C/OL]//2009 IEEE International Conference on Communications. 2009: 1-6. DOI: 10.1109/ICC.2009.5199557.

[15] Sun G L, Xue Y, Dong Y, et al. An novel hybrid method for effectively classifying encrypted traffic[C/OL]//2010 IEEE Global Telecommunications Conference GLOBECOM 2010. 2010: 1-5. DOI: 10.1109/GLOCOM.2010.5683649.

[16] Gu R, Wang H, Ji Y. Early traffic identification using bayesian networks[C/OL]//2010 2nd IEEE InternationalConference on Network Infrastructure and Digital Content. 2010: 564-568. DOI: 10.1109/ICNIDC.2010.5657833.

[17] Tabatabaei T S, Karray F, Kamel M. Early internet traffic recognition based on machine learning methods[C/OL]//2012 25th IEEE Canadian Conference on Electrical and Computer Engineering (CCECE). 2012: 1-5. DOI: 10.1109/CCECE.2012.6335034.

[18] Huijun C, Hong S, Hong Z. Early recognition of internet service flow[C/OL]//2013 22nd Wireless and Optical Communication Conference. 2013: 464-468. DOI: 10.1109/WOCC.2013.6676412.

[19] Draper-Gil G, Lashkari A H, Mamun M S I, et al. Characterization of encrypted and vpn traffic using time-related features[C]//ICISSP. 2016.

[20] Yamansavascilar B, Guvensan M A, Yavuz A G, et al. Application identification via network traffic classification[C/OL]//2017 International Conference on Computing, Networking and Communications (ICNC). 2017: 843-848. DOI: 10.1109/ICCNC.2017.7876241.

[21] Lashkari A H, Draper-Gil G, Mamun M S I, et al. Characterization of tor traffic using time based features[C]//ICISSP. 2017.

[22] Shahbar K, Zincir-Heywood A N. How far can we push flow analysis to identify encrypted anonymity network traffic?[C/OL]//NOMS 2018 - 2018 IEEE/IFIP Network Operations and Management Symposium. 2018: 1-6. DOI: 10.1109/NOMS.2018.8406156.

[23] Shahbar K, Zincir-Heywood A N. Packet momentum for identification of anonymity networks [C/OL]//Journal of Cyber Security and Mobility: volume 6. 2017: 27-56. DOI: https://doi.org/10.13052/jcsm2245-1439.612.

[24] Wang Z. The applications of deep learning on traffic identification[C]//Black Hat USA 2015. 2015.

[25] Lotfollahi M, Siavoshani M J, Zade R S H, et al. Deep packet: A novel approach for encrypted traffic classification using deep learning[J]. Soft Computing, 2020, 24(3): 1999-2012.

[26] Wang W, Zhu M, Wang J, et al. End-to-end encrypted traffic classification with one-dimensional convolution neural networks[C]//2017 IEEE International Conference on Intelligence and Security Informatics (ISI). IEEE, 2017: 43-48.

[27] Wang W, Zhu M, Zeng X, et al. Malware traffic classification using convolutional neural network for representation learning[C]//2017 International Conference on Information Networking (ICOIN). IEEE, 2017: 712-717.

[28] Lim H K, Kim J B, Heo J S, et al. Packet-based network traffic classification using deep learning [C/OL]//2019 International Conference on Artificial Intelligence in Information and Communication (ICAIIC). 2019: 046-051. DOI: 10.1109/ICAIIC.2019.8669045.

[29] He K, Zhang X, Ren S, et al. Deep residual learning for image recognition[C]//Proceedings of the IEEE conference on computer vision and pattern recognition. 2016: 770-778.

[30] Rezaei S, Liu X. How to achieve high classification accuracy with just a few labels: A semi-supervised approach using sampled packets[J]. ArXiv, 2019, abs/1812.09761.

[31] Zhang J, Li F, Ye F, et al. Autonomous unknown-application filtering and labeling for dl-based traffic classifier update[C/OL]//IEEE INFOCOM 2020 - IEEE Conference on Computer Communications. 2020: 397-405. DOI: 10.1109/INFOCOM41043.2020.9155292.

[32] 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): 8.

[33] Lopez-Martin M, Carro B, Sanchez-Esguevillas A, et al. Network traffic classifier with convolutional and recurrent neural networks for internet of things[J/OL]. IEEE Access, 2017, 5: 18042-18050. DOI: 10.1109/ACCESS.2017.2747560.

[34] Liu C, He L, Xiong G, et al. Fs-net: A flow sequence network for encrypted traffic classification[C/OL]//IEEE INFOCOM 2019 - IEEE Conference on Computer Communications. 2019: 1171-1179. DOI: 10.1109/INFOCOM.2019.8737507.

[35] Wang X, Chen S, Su J. App-net: A hybrid neural network for encrypted mobile traffic classification[C/OL]//IEEE INFOCOM 2020 - IEEE Conference on Computer Communications Workshops (INFOCOM WKSHPS). 2020: 424-429. DOI: 10.1109/INFOCOMWKSHPS50562.2020.9162891.

[36] Wang M, Zheng K, Luo D, et al. An encrypted traffic classification framework based on convolutional neural networks and stacked autoencoders[C/OL]//2020 IEEE 6th International Conference on Computer and Communications (ICCC). 2020: 634-641. DOI: 10.1109/ICCC51575.2020.9344978.

[37] Goodfellow I, Pouget-Abadie J, Mirza M, et al. Generative adversarial nets[J]. Advances in neural information processing systems, 2014, 27.

[38] Wang P, Li S, Ye F, et al. Packetcgan: Exploratory study of class imbalance for encrypted traffic classification using cgan[C/OL]//ICC 2020 - 2020 IEEE International Conference on Communications (ICC). 2020: 1-7. DOI: 10.1109/ICC40277.2020.9148946.

[39] Mirza M, Osindero S. Conditional generative adversarial nets[J]. arXiv preprint arXiv:1411.1784, 2014.

[40] Japkowicz N, et al. Learning from imbalanced data sets: a comparison of various strategies[C]//

AAAI workshop on learning from imbalanced data sets: volume 68. AAAI Press Menlo Park, CA, 2000: 10-15.

[41] 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.

[42] Guo Y, Xiong G, Li Z, et al. Combating imbalance in network traffic classification using gan based oversampling[C/OL]//2021 IFIP Networking Conference (IFIP Networking). 2021: 1-9. DOI: 10.23919/IFIPNetworking52078.2021.9472777.

[43] Li C, Xu T, Zhu J, et al. Triple generative adversarial nets[J]. Advances in neural information processing systems, 2017, 30.

[44] Aouedi O, Piamrat K, Bagadthey D. A semi-supervised stacked autoencoder approach for network traffic classification[C/OL]//2020 IEEE 28th International Conference on Network Protocols (ICNP). 2020: 1-6. DOI: 10.1109/ICNP49622.2020.9259390.

[45] Xing J, Wu C. Detecting anomalies in encrypted traffic via deep dictionary learning[C/OL]// IEEE INFOCOM 2020 - IEEE Conference on Computer Communications Workshops (INFOCOM WKSHPS). 2020: 734-739. DOI: 10.1109/INFOCOMWKSHPS50562.2020.9162940.

[46] Van den Oord A, Li Y, Vinyals O. Representation learning with contrastive predictive coding [J]. arXiv e-prints, 2018: arXiv-1807.

[47] He K, Fan H, Wu Y, et al. Momentum contrast for unsupervised visual representation learning[C]//Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. 2020: 9729-9738.

[48] Chen T, Kornblith S, Norouzi M, et al. A simple framework for contrastive learning of visual representations[C]//International conference on machine learning. PMLR, 2020: 1597-1607.

[49] Chen X, Fan H, Girshick R, et al. Improved baselines with momentum contrastive learning[J]. arXiv preprint arXiv:2003.04297, 2020.

[50] Chen T, Kornblith S, Swersky K, et al. Big self-supervised models are strong semi-supervised learners[J]. Advances in neural information processing systems, 2020, 33: 22243-22255.

[51] Le-Khac P H, Healy G, Smeaton A F. Contrastive representation learning: A framework and review[J/OL]. IEEE Access, 2020, 8: 193907-193934. DOI: 10.1109/ACCESS.2020.3031549.

[52] Yang Y, Kang C, Gou G, et al. Tls/ssl encrypted traffic classification with autoencoder and convolutional neural network[C/OL]//2018 IEEE 20th International Conference on High Performance Computing and Communications; IEEE 16th International Conference on Smart City; IEEE 4th International Conference on Data Science and Systems (HPCC/SmartCity/DSS). 2018: 362-369. DOI: 10.1109/HPCC/SmartCity/DSS.2018.00079.

[53] Wei J, Zou K. Eda: Easy data augmentation techniques for boosting performance on text classification tasks[J]. ArXiv, 2019, abs/1901.11196.

[54] Glorot X, Bordes A, Bengio Y. Deep sparse rectifier neural networks[C]//Proceedings of the fourteenth international conference on artificial intelligence and statistics. JMLR Workshop and Conference Proceedings, 2011: 315-323.

[55] Gutmann M, Hyvärinen A. Noise-contrastive estimation: A new estimation principle for unnormalized statistical models[C]//Proceedings of the thirteenth international conference on artificial

intelligence and statistics. JMLR Workshop and Conference Proceedings, 2010: 297-304.

[56] Lamping U, Warnicke E. Wireshark user's guide[J]. Interface, 2004, 4(6): 1.

[57] Kingma D P, Ba J. Adam: A method for stochastic optimization[J]. arXiv preprint arXiv:1412.6980, 2014.

[58] Littlechild S C, Owen G. A simple expression for the shapley value in a special case[J]. Management Science, 1973, 20(3): 370-372.

[59] Zhang Q s, Zhu S C. Visual interpretability for deep learning: a survey[J]. Frontiers of Information Technology & Electronic Engineering, 2018, 19(1): 27-39.

[60] Choo J, Liu S. Visual analytics for explainable deep learning[J/OL]. IEEE Computer Graphics and Applications, 2018, 38(4): 84-92. DOI: 10.1109/MCG.2018.042731661.

[61] Wang T, Isola P. Understanding contrastive representation learning through alignment and uniformity on the hypersphere[C]//International Conference on Machine Learning. PMLR, 2020: 9929-9939.

[62] Devlin J, Chang M W, Lee K, et al. Bert: Pre-training of deep bidirectional transformers for language understanding[J]. arXiv preprint arXiv:1810.04805, 2018.

[63] Liu Y, Ott M, Goyal N, et al. Roberta: A robustly optimized bert pretraining approach[J]. arXiv preprint arXiv:1907.11692, 2019.

[64] Lan Z, Chen M, Goodman S, et al. Albert: A lite bert for self-supervised learning of language representations[J]. arXiv preprint arXiv:1909.11942, 2019.

[65] Yang Z, Dai Z, Yang Y, et al. Xlnet: Generalized autoregressive pretraining for language understanding[J]. Advances in neural information processing systems, 2019, 32.

[66] Song K, Tan X, Qin T, et al. Mass: Masked sequence to sequence pre-training for language generation[J]. arXiv preprint arXiv:1905.02450, 2019.

[67] Dong L, Yang N, Wang W, et al. Unified language model pre-training for natural language understanding and generation[J]. Advances in Neural Information Processing Systems, 2019, 32.

[68] Lin T Y, Goyal P, Girshick R, et al. Focal loss for dense object detection[C]//Proceedings of the IEEE international conference on computer vision. 2017: 2980-2988.

[69] Prechelt L. Early stopping-but when?[M]//Neural Networks: Tricks of the trade. Springer, 1998: 55-69.

[70] Srivastava N, Hinton G, Krizhevsky A, et al. Dropout: a simple way to prevent neural networks from overfitting[J]. The journal of machine learning research, 2014, 15(1): 1929-1958.

[71] Gao T, Yao X, Chen D. Simcse: Simple contrastive learning of sentence embeddings[J]. arXiv preprint arXiv:2104.08821, 2021.

[72] Konečnỳ J, McMahan B, Ramage D. Federated optimization: Distributed optimization beyond the datacenter[J]. arXiv preprint arXiv:1511.03575, 2015.

[73] McMahan B, Moore E, Ramage D, et al. Communication-efficient learning of deep networks from decentralized data[C]//Artificial intelligence and statistics. PMLR, 2017: 1273-1282.

[74] Zhao Y, Li M, Lai L, et al. Federated learning with non-iid data[J]. arXiv preprint

arXiv:1806.00582, 2018.

[75] Li X, Huang K, Yang W, et al. On the convergence of fedavg on non-iid data[J]. arXiv preprint arXiv:1907.02189, 2019.

[76] WHDY. Fedavg[EB/OL]. 2022[2022-04-07]. https://github.com/WHDY/FedAvg.