《基于卷积神经网络（CNN）的网络流量分类》优秀本科毕设实验总结

原创

从毕业设计完成到现在不知不觉已经大半年了，依稀记得当时毕设双选时给老师发邮件时的忐忑，老师给我答复时的喜悦。在毕业设计的过程中，遇到过许多困难。刚开始看文献综述的时候都是陌生的概念，寸步难行，面临难题时有时候想了很长时间但是就是想不出来解决方案，但是等到熬过概念输入的时期，耐心分析出问题的原因时那种豁然开朗的感觉真的很美妙。在研究的期间，我收获的不仅仅是专业的知识，更宝贵的是遇到问题的处理方法，怎么发现问题，分析问题，解决问题，总结问题。如何保持主动积极的状态，清楚的表达自己的想法，更加有效和别人沟通，做出严谨科学的解决方案，这些都是原来我不曾注重的。

还记得每周一次的组会，有时候有小的突破就会很期待周二的组会，把自己的发现和结果汇报分享，有时候卡住了没进度就会很慌，哈哈。不过只要用心总是会有结果的，最后预答辩时候在院长和主任面前展示自己的成果，回答老师的问题，那种扎扎实实不怕问的感觉真的很舒服。

当院长最后答辩时候说你们评上优秀毕设的都是咱计算机学院的小牌面，以后出去了要好好做事，认真工作，把自己的技术搞好，不要给学校丢人。听到这些话还是很激动的，为自己的毕业画上了一个不错的句号，未来也要继续加油！
今天周日适合写一些总结的文章，那就复盘一下我的小宝贝吧。
实验有一部分都是看文献的学习复现，所以就不怎么写了，只写写自己思考了很多，做了很多的部分吧。

目录

背景

近年来，随着互联网的快速发展，各种新型的互联网应用接踵而至，网络规模不断扩大，网络流量也日益繁多。网络流量分类技术作为网络管理中的关键手段之一，不仅可以承担分辨加密流量以及恶意流量确保网络安全的任务，还可以为网络管理者提供可靠的网络资源使用情况，以便网络管理者进行合理的资源分配以及科学的调整网络架构。基于端口，深度包检测和经典的机器学习方法等传统网络流量分类技术已被广泛使用，但是由于互联网的急剧变化，特别是加密流量的大大增加，这些方法的准确率已经下降。鉴于深度学习方法在图像分类领域的优良表现，网络流量分类研究学者开始关注于基于深度学习的网络流量分类技术，并且做出了许多尝试，取得了很好的效果。

本次研究中使用卷积神经网络作为流量分类的模型。首先对原始流量数据进行预处理，然后使用深度学习框架Tensorflow完成模型的搭建，接着输入标准数据实现对混合流量分类，模型不仅分辨出10种常规流量以及10种加密流量，还可以分辨出8种常规流量以及2种恶意流量。在此之后，面对数据不平衡问题，在数据预处理阶段本文尝试着从采样角度，交叉验证角度来解决这个问题，做了多组对比实验，F1得分和准确率在原来的基础上都提升了2个百分点。最后对实验数据和结果整理分析，思考此次研究的得失，发现未来可以继续改进的方向。

一、实验基础

1.1 两个公开数据集：ISCXVPN2016数据集和USTC-TFC2016数据集
实验采用的深度学习框架为tensorflow2.0，模型为CNN。
1.2 硬件平台：
操作系统 Window10
处理器 AMD A10-8700P
内存 8192MB RAM
显卡 AMD Radeon R6 Graphics
硬盘 500G三星固态
1.3 软件环境：
Anaconda3，python 3.7， Tensorflow 2.1.0，numpy 1.18.1，imageio 2.6.1，opencv3.4.5.20等。
1.4 评价标准：

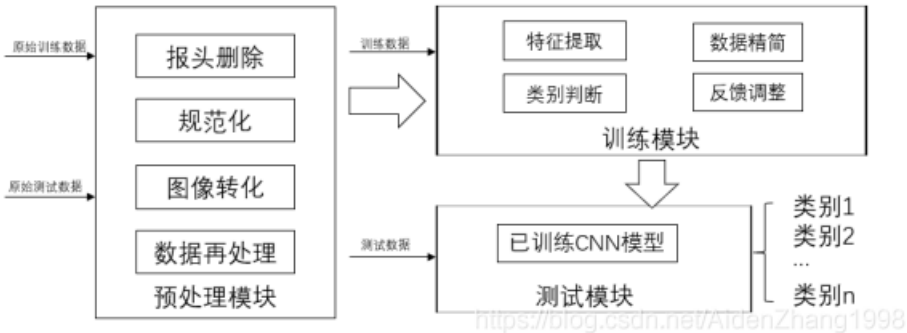
	分类结果	真实标签
TP	正例	正例
FP	正例	负例
TN	负例	负例
FN	负例	正例

最终评价指标为准确率（Accuracy，ACC），精准度（Precision），召回率（Recall）以及F1得分（F1-score）。

$$ACC = \frac{TP+TN}{TN+FN+TP+FP} \times 100\%$$
$$Precision = \frac{TP}{FN+TP} \times 100\%$$
$$Recall = \frac{TP}{TP+FP} \times 100\%$$
$$F1-score = 2 \times \frac{Precision \times Recall}{Precision+Recall} \times 100\%$$

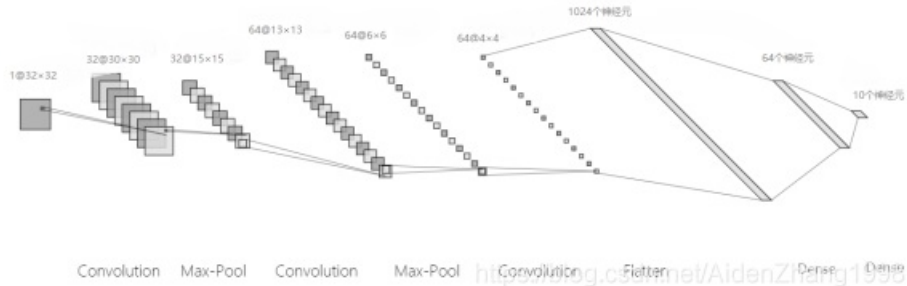
二、系统设计

2.1 系统框架



流量数据输入到预处理模块中，经过报头删除，规范化，图像转化以及数据再处理四个步骤输出为卷积神经网络可以直接使用的数据。对于训练数据，将它输入到训练模块中，经过特征提取，数据精简，类别判断以及反馈调整训练出卷积神经网络模型。对于测试数据，将它输入到含有训练完成的卷积神经模型的测试模块，根据分类结果，完成对系统的评估。

2.2 卷积神经网络模型



整个模型共有八个层次。依次是卷积层，池化层，卷积层，池化层，卷积层，展平层，全连接层，全连接层。
第一个卷积层输入数据为32×32×1，代表一个预处理模块处理好的灰度图片。共有32个卷积核，大小为3×3，x轴和y轴步长都为1，非填充模式，结果使用ReLU激活函数进行转化，生成32个30×30的平面。
第一个池化层接收上层数据，为最大池化层，池化层大小为2×2，x轴和y轴步长都为2，生成32个15×15的矩阵。
第二个卷积层接收上层数据，共64个卷积核，大小为 3×3， x轴和y轴步长都为1，结果进行ReLU函数激活，生成64个13×13的矩阵。
第二个池化层接收上层数据，为最大池化层，池化层的大小为2×2，x轴和y轴的步长都为2，生成64个6×6的矩阵。
第三个卷积层接收上层数据，共64个卷积核，大小为 3×3， x轴和y轴步长都为1，结果进行ReLU函数激活，生成64个4×4的矩阵。
展平层接受上层数据，将64个矩阵展平降维，输出为1024个神经元。
第一个全连接层接受展平层的输出数据，处理为64个神经元。
末尾的层是全连接层，使用10个神经元完成10分类。

三、样本不平衡问题（自己的思考很多）

3.1 实验设置

第一部分实验为基础五组实验，目的是调整卷积神经网络的模型架构以及其他实验设置，使得系统能够正常分类出10种常规流量，10种VPN加密流量以及2种恶意流量。并且使这五组实验的分类准确率达到90%以上。
在第一部分实验结束后在10种常规流量中分类实验中遇到了**数据不平衡问题**，也是本文要解决的问题。例如在此次实验中aim流量样本数为4099个，icq流量样本数为3476个，而其他8种流量样本十分充足，采样到了5000个。在网络训练过程中，少数类别样本个数较少，模型对这些类别特征没有充分学习，所以会导致误判，结果就是本来是少数类别的流量会判给其他类别，总体分类的准确率，精度，召回率等都会有一定程度的降低，在接下来的两部分实验中，本文就是要解决这个问题，从数据不平衡的角度提高分类的各个指标。
第二部分为采样三组实验，在第一部分实验确定的模型架构等基础之上，针对数据不平衡问题，从采样的角度来在数据预处理模块做出改变，测试3种采样方案对准确率的影响。
第三部分为交叉验证三组实验，面对数据不平衡问题，从数据充分利用的角度，借鉴交叉验证的思想，测试修改的五折交叉方法和十折交叉方法对准确率的影响。

3.2 采样三组实验

本次实验为了研究不平衡数据的不同处理方法对准确率的影响，先在ISCXVPN2016数据集选取10个非VPN流量类别，从样本采样的角度来进行实验，并对3种采样方法对比：仅过采样，仅欠采样，过采样和欠采样结合。
3种方法目的是一样的，即使得各个流量类别的训练样本数目保持一致。仅过采样方法是样本数目都处理为样本最多的流量类别的数目。仅欠采样方法是样本数目都处理为样本最少的流量类别的数目。过采样和欠采样结合方法是在样本最多的流量类别的数目和样本最少的流量类别的数目之间探索出一个合适的样本数目。
对于第一种仅过采样，在第一部分实验中50000份数据就可以达到98%的准确率，所以实际上没有必要采取那么多的样本，一定量的样本是可以满足模型的训练学习的，而且对于视频流量（Netflix，Youtobe等）的样本数目可能是几百万，对于小类就要复制几百次，复制次数到达一定程度时也是对于模型分类的提高没有多大的效果的。在此次研究中，作者的硬件设施也是不支持的，所以综合分析，这个方法被排除了。

3.2.1 原始对照实验

对于第一种实验不做任何处理，作为对照实验，与基础五组实验的非VPN10分类实验保持一致。

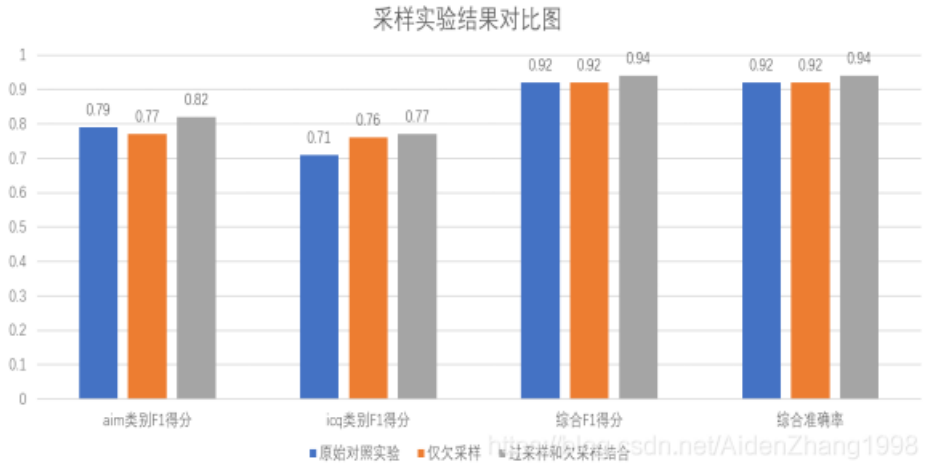
3.2.2 仅欠采样

在仅欠采样实验中，保证每个类别的训练集和验证集样本数目一致，每类样本数目采取至最小类别流量的样本数目，最小类别流量为3476个。其他条件与原始对照实验保持一致。

3.2.3 过采样和欠采样结合

在过采样和欠采样结合实验中，每个类别的训练集和验证集样本数目也是一样的，按照训练集，验证集，测试集的比例为8:1:1的原则。那么每个类别的训练集和验证集共有4500个样本，对大类流量采取样本至4500个，对小类流量（aim，icq）采用随机复制扩充至4500个样本。测试集不做处理，保证数据的原始分布。其他条件与原始对照实验保持一致。

在实验结果分析时，抽取前三组每个实验的aim类别流量F1得分，icq类别流量F1得分,综合F1得分，综合准确率四个指标进行比较。如图为前三组采样对比实验结果图。



仅欠采样实验与原始对照实验相比相差并不大，虽然保证了样本数目一致，但是各项指标没有明显的增加，原因是仅欠采样的样本数目减少，影响了特征的充分学习。过采样和欠采样结合的实验各项指标都有所增加，综合F1得分和综合准确率增加了两个百分点。本次的实验确定了最优的采样方法，即过采样和欠采样结合的方法。

3.3 交叉验证三组实验

本次验证交叉验证方法借鉴经典交叉验证思想，但是针对本次实验又加以改变，使得分类系统可以充分利用数据集，减少数据不平衡时对准确率的影响，提高准确率的同时避免过拟合。数据集选取10个非VPN流量类别，根据对照实验处理为三组，第一组为训练集，验证集和测试集大致为8:1:1的固定的数据集，第二组实验数据集中首先拿出十分之一的数据集保留为测试集，然后其余部分为训练集和验证集，训练集和验证集遵循五折交叉验证的原则，测试集仍然保证对于模型是未知的。第三组实验数据集中同第二组相比训练集和验证集遵循十折交叉验证的原则，其他与第二组保持一致。

validation	train	train	train	train	test
train	**validation**	train	train	train	test
train	train	**validation**	train	train	test
train	train	train	**validation**	train	test
train	train	train	train	**validation**	test

这样就可以对小类数据充分的利用，而且对于模型首先划出的测试集也是未知的，提高准确率的同时也保证模型的可泛化性，避免过拟合问题。

3.3.1 原始对照实验

对于第一个实验不做任何处理，作为对照实验，与基础五组实验的非VPN10分类实验保持一致。

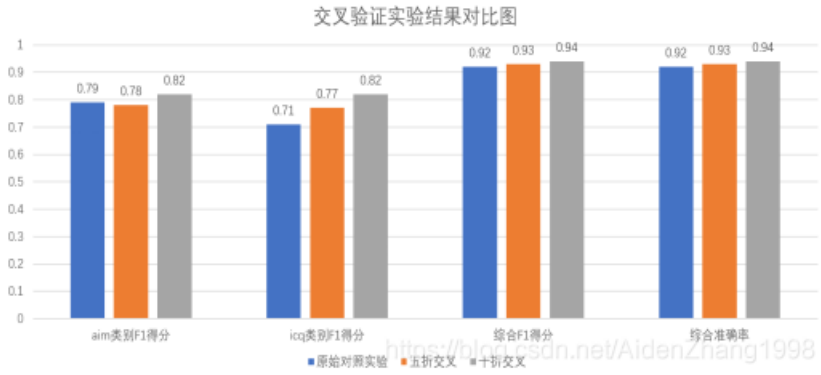
3.3.2 五折交叉实验

五折交叉实验的改变在数据输入上，首先拿出十分之一的数据作为测试集，测试集不参与模型的训练，保证对于模型它是未知的。然后对剩下的五分之四（训练集和验证集）做交叉验证，每轮10次迭代，继承上一轮的模型，共50次迭代。与对照实验的50次迭代一致，其他条件和第一个对照实验保持一致。

3.3.3 十折交叉实验

对于十折交叉实验，首先拿出十分之一的数据保留为测试集，然后对剩下的十分之九（训练集和验证集）做交叉验证，每轮5次迭代，继承上一轮的模型，共50次迭代，总的与对照实验一致，其他条件和第一个对照实验保持一致。

在实验结果分析时，抽取三组每个实验的aim类别流量F1得分，icq类别流量F1得分，综合F1得分，综合准确率四个指标进行比较。如图为三组对比交叉实验结果图。



十折交叉实验各个指标值都超过了五折交叉实验的指标值，十折交叉实验相比于原始对照实验，aim类别流量F1得分提升了3个百分点，icq类别F1得分提升了11个百分点，综合得分提升了2个百分点，综合准确率提升了2个百分点。实验证明了提出的改进交叉验证的有效性。

3.4 实验总结（思考分析结果，猜想一些可能的原因）

在本次研究中，首先做了基础的分类实验，模型可以分出10种常规流量和对应的10种VPN加密流量，还有8种常规流量和2种恶意流量，并且分类准确率可以达到90%以上。然后从采样角度解决数据不均衡问题，通过几个方法的比较，最终的过采样和欠采样结合方法比未经采样处理原始实验的F1得分和准确率都提升了2个百分点。最后从数据充分利用的角度尝试了五折交叉实验和十折交叉实验，最优的十折交叉方法相对于原始对照实验的F1得分和准确率也是都提升了2个百分点，显示了此方法的可靠性。
实际上在实验中还有一个有趣的地方，就是在仅欠采样的实验中，每个类别的样本数目都为3476个，但是小类流量（aim，icq）的F1得分在80%左右，其他大类流量F1得分都在90%以上。在其他实验中，小类F1得分在各种处理下都有所提升，但是同样还是和大类有着显著差异。
为什么样本个数一样的同时小类分类效果依然和大类有这么大的差别呢？数据充分利用之后还是不能完全弥补它们的差距呢？
后来本文提出了一个可能的原因，可能因为小类别流量不仅仅是数据包数目少，每个数据包里面的数据内容也是很少的，在数据预处理模块本文选取特征的手段是原始数据包数据，那么从一开始小类别流量和大类别流量在样本数目和样本内容上都有着显著的差异，仅仅弥补样本数目是不够的。
回过头来观察小类图片和大类图片，发现确实小类的图片大部分都是黑色居多，大类图片大部分都是白色居多。这个现象是因为在数据预处理阶段取数据包的前1024个字节，不够的进行零填充，小类的数据包内容很少，大都进行了零填充，0在灰度图片意味着黑色。于是这就验证了本文的猜想，使用原始数据包数据作为特征时，数据不平衡的问题要分解为两个问题，一个是样本数目，一个是样本内容。但是样本内容是一开始就确定了的，无法改变。使用原始数据包数据作为特征面对数据不平衡问题时，有着无法弥补的先天缺陷。
本文所做的采样和交叉验证实验从样本个数的角度出发，但是样本内容是无法改变的，要想改变样本内容，就要更换选取特征的手段，不再使用原始数据包数据作为特征，而是那些对于大类和小类没有显著差异的特征，这样新的特征没有容量上的差异，再加上从样本个数角度的处理，相信更进一步的解决数据不平衡问题。不过由于时间原因，此次研究中没有再进行探索。

四、展望（下一步的可能，很珍贵！）

在面对数据不平衡问题时，本文尝试了采样方法和改进的交叉验证方法，取得了一定的效果，但是小类流量的准确率和F1得分还是和大类有着差距，经过思考和查阅资料，提出了几个可以尝试的方向。
选取时间序列等流特征代替数据包数据作为数据来源。在此次研究中，通过实验发现小类流量的差异和大类流量的差异在两个方面，一个是数据包的内容，一个是数据包的数目。如果选取数据来源时使用大类流量和小类流量没有差异的流特征，那么就避开了小类流量和大类流量的样本内容的差异，只剩下了样本的数目差异。这个方法的关键在于选取哪些小类流量和大类流量差异不大的流特征。
对于小类流量使用更加严格的惩罚因子。面对小类流量误判犯错时，给予它相比于大类流量更加严格的惩罚，加快小类流量的学习进程[7]。这个需要对神经网络有着深入的了解。
对于小类流量加入人工构造数据。例如SMOTE(Synthetic Minority Over-sampling Technique)。SMOTE是过采样算法，它可以人工生成数据集中从未出现过新的小类样本。具体就是使用K近邻在少数类样本之间插值生成新的样本。这个方法难点在需要一些工作找寻合适的K值。
加入机器学习算法。在这个方面前人做出了尝试，例如V. Tong等人提出了一种基于卷积神经网络、基于NetFlow和基于数据包特征的分类方法。第一阶段使用随机森林检测两种流量。第二阶段将第一阶段剩下的流量采用卷积神经网络等手段分为其他三种流量。最后的实验结果表明可以分类出五种基于QUIC的流量，准确度很高（约99%）。
不过这种办法还是很耗时耗力的。在实践中会遇到几个比较难的问题，第一是首先要找到对小类别流量数据友好的机器学习算法，这中间要尝试很多机器学习算法，要做很多工作，第二是找到合适的算法之后，面对数据集的迁移变化，可能这个合适的算法就会变得不再适用，接着又回到了第一个问题，所以还是挺耗费时间和精力的，但是一个不错的尝试方向。

五、相关文档

工程代码以及相关文档已经上传至github。