分类号 TP311　　密 级

U D C　　编 号 10486

武漢大學

硕 士 学 位 论 文

基于深度神经网络的加密流量分类

研 究 生 姓 名 ：万晶

学　　号 ：2016202110030

指导教师姓名、职称 ：吴黎兵 教授

专 业 名 称 ：计算机软件与理论

研 究 方 向 ：流量分类

二〇一九年五月

MASTER'S DEGREE THESIS OF
WUHAN UNIVERSITY

A Classification Method of Encrypted Traffic based on Deep Neural Network

By
Jing Wan

May 2019

论文原创性声明

本人郑重声明：所呈交的学位论文，是本人在导师指导下，独立进行研究工作所取得的研究成果。除文中已经标明引用的内容外，本论文不包含任何其他个人或集体已经发表或撰写过的研究成果。对本文的研究做出贡献的个人和集体，均已在文中以明确方式标明。本声明的法律结果由本人承担。

学位论文作者（签名）：

年　　月　　日

摘要

随着用户网络安全意识的提升，大量的应用程序使用加密隧道来传输数据，如 VPN，SSH 等。网络中的流量种类变得更加多样，当前网络流量中不仅含有传统的数字，图像，流媒体等，还包含了多种加密流量。由于缺少对加密流量的了解以及有效的识别手段，加密流量提升了网络管理的难度，对网络安全构成了极大的威胁。对加密网络流量的分类有助于了解网络流量的组成，有利于网络管理和网络安全。

当前，随着网络流量加密技术的广泛运用，传统的网络流量分类方法已经不适用于当今复杂多变的网络。本文分析了由 VPN 加密并传输的流量并对其分类方法进行了探索。

本文提取了加密网络流量基于时序的特征，构建了基于深度神经网络的加密流量分类模型，识别并分类了 7 种不同类别的加密网络流量，并与常用的朴素贝叶斯分类算法进行对比。同时，本文还在小范围内研究了 mini-batch size 的取值是否会影响深度神经网络的分类效果，探讨 mini-batch size 的变化对深度神经网络分类效果的影响。

实验结果显示，基于深度神经网络的加密流量分类模型的分类能力远好于朴素贝叶斯方法。在训练时，mini-batch size 会对深度神经网络模型产生不同程度的影响，当 mini-batch size 为 40 时，深度神经网络模型的分类能力最好。

关键词：加密流量分类；深度神经网络；深度学习；SSL/TLS

ABSTRACT

As user network security awareness increases, a large number of applications use encrypted tunnels to transmit data, such as VPN, SSH, and so on. The types of traffic in the network become more diverse. The current network traffic contains not only traditional digital, image, streaming media, but also a variety of encrypted traffic. Due to the lack of understanding of encrypted traffic and effective identification methods, encrypted traffic increases the difficulty of network management and poses a great threat to network security. The classification of encrypted traffic helps to understand the composition of network traffic, which is beneficial to network management and network security.

With the widespread use of network traffic encryption technology, traditional traffic classification methods have gradually failed. This paper analyzes the traffic encrypted and transmitted by VPN and explores its classification method.

By extracting the time series characteristics of encrypted traffic, this paper uses the classification model of deep neural network to classify the traffic of seven different categories in the encrypted traffic, and compares it with the commonly used naive Bayesian classification algorithm. At the same time, this paper also studies the batch size that affects the training of deep neural network models, and discusses the effect of batch size on the classification of deep neural network models.

Experiments show that the classification ability of encrypted traffic classification model based on deep neural network is much better than the naive Bayesian method. During training, the batch size has different effects on the deep neural network model. When the batch size is 40, the deep neural network model has the best classification ability.

Key words: Encrypted traffic classification; deep neural network; deep learning; SSL/TLS

目录

第 1 章 绪论

1.1 研究背景和意义

随着计算机网络技术的发展，互联网已逐渐深入人们生活的各个方面，极大的改变了人们的生活方式。2019 年 2 月 28 日，《中国互联网发展报告 2019》[1]发布。报告显示，截至 2018 年 12 月，中国网民规模达 8.29 亿，网民以 10~39 岁群体为主，人均周上网 27.6 小时，互联网成为人们生活不可缺失的部分。

互联网的快速发展与应用，让人们的沟通的更加快捷，丰富了人们的娱乐生活，方便了人们的生活。2017 年 12 月，《互联网趋势报告（2017-2018）》[2]发布，报告指出，互联网已成为全球经济增长主要推动力，用户与流量不断扩大。全球移动市场中，游戏类、生活服务、电子商务、主体和办公学习类应用为全球移动应用市场应用规模前五位。

随着各种类型的 Web 网络应用数量的不断增多，Internet 已经渗透到社交生活的各个方面。除了传统的 HTTP、FTP、Email 等应用外，新的应用如流媒体、P2P[3]、VoIP、网络游戏、VR 等不断涌现。同时，网络攻击、网络病毒、钓鱼网站，非法信息的传播，对网络的安全与管理造成了巨大威胁。

与此同时，互联网的飞速发展让网络安全迎来了巨大的挑战。互联网的开放性，不仅使网络中的流量更加多样化，也让各种网络攻击层出不穷。大量的网络流量，也使网络资源的分配变得更加困难。在线用户的激增使得管理与治理公共舆论变得更加困难。《中国互联网站发展状况及其安全报告（2018）》[4]指出，仅 2017 年，我国境内被篡改的网站数量为 20111 个，国家信息安全漏洞共享平台（CNVD）共新增收录通用软硬件漏洞 15955 个，我国境内的 29236 个网站被植入后门。

随着用户网络安全意识的提升，大量的应用程序使用加密流量来传输数据，网络中的流量种类变得更加多样，当前网络流量中不仅含有传统的数字，图像，文件，流媒体等，还包含了多种的加密流量。由于缺少对加密网络流量的了解以及有效的识别手段，加密流量提升了网络管理的难度，对网络安全构成了极大的威胁。对加密流量的分类的研究是十分有必要的，只有了解网络流量的组成，才能更好的管理网络。

本文将构建一个基于深度神经网络的加密流量分类模型，对通过 VPN 隧道传输的七种网络流量分类。基于深度神经网络的加密流量分类模型通过提取加密网络流量的基于时序的统计特征，构建深度神经网络分类模型。实验结果表明，基于深度神经网络的加密流量分类模型能很好的对加密网络流量分类，并且，通过对比，基于深度神经网络的加密流量分类模型分类效果远好于朴素贝叶斯分类

算法。本文还研究了影响深度神经网络优化的因素 mini-batch size，通过对比实验，发现当 mini-batch size 的大小为 45 时，深度神经网络模型的分类效果最优。

1.2 国内外研究现状

近年来，网络流量的分类变得非常重要，因为对于网络运营商来说，拥有流经其网络的信息是非常有用的。这些信息的一些用途是：对可能需要服务质量（QoS）的应用程序包进行优先级排序，或禁用正在进行非法行为的用户的网络访问。目前该领域的几个问题是：网络中立性、在网络上共享受版权保护的内容以及恶意用户和网络管理员之间的冲突。

随着隐藏数据标识信息的技术的发展，网络流量分类技术已经被使用了很多年，并且要求越来越复杂。最初的方法是使用与 TCP 连接关联的端口号来标识网络流量。这是一种非常快速的方法，使用众所周知的端口号。这种方法有几个缺点。此外，某些应用程序能够协商一个动态端口号来交换数据，从而避免使用标准端口号。最近，该技术已逐渐失效。

深度数据包检查通过检查数据包的内容对网络流量进行分类。为了实现这一点，将数据包的内容与已知应用程序的存储模式进行比较。这种方法可以得到精确的结果，但是计算代价高。这种技术也很容易被加密技术所回避，这使得直接查看数据成为不可能。另一种方法是识别用户和服务器之间的交互模式。应用程序以已知的方式与服务器交互，这可以用来识别应用程序，称为基于主机行为的方法。

流量分类是一个具有挑战性的研究方向。为了解决基于端口分类的问题，之前的工作主要关注基于非端口的方法来识别网络流量，包括基于负载的分析、行为分析和聚类分析。基于有效负荷的方法（例如，[5，6]）是标准方法。这种方法尝试将包有效负载与签名库匹配，签名库由与特定攻击或应用程序关联的惟一字节序列组成。除了在监视加密的或包采样的流量时失败之外，基于有效负载的方法的另一个缺点是字节序列通常不是特定流量类型的惟一，这导致了 NIDS 中众所周知的错误警报问题。基于行为特征（如[7，8]）的分类方法侧重于建立传输层度量（如连接持续时间和包大小）的统计模型，以区分应用程序。这些方法局限于一组以前观察到的行为，通常内置在分类器中，在这些部署情况下，由于缺乏信息，这些方法无法监视所有的应用程序流量，因此挑战极大。基于集群的方法，如[9，10，11]，通过使用标准的机器学习方法，根据传输层特征的相似性将流量划分为组。这些方法也常常应用受限，因为它们需要有标记良好的训练集来学习，因此没有自动适应新协议的方法。虽然这些方法有其优点，但最终受限于所使用的议定书所提供的信息多样性的限制。而使用上述方法进行流量分类可

能很有用，但它们常常忽略诊断和纠正问题所需的关键细节，并且可能永远无法准确地完全区分所有流量类型。因此，有必要从更广泛的角度来看待问题。

1.2.1 基于标准端口匹配

对于初期的应用，互联网号码分配机构 IANA[12]都规划了专门的 TCP/UDP 服务端口。IANA 最初是以先到先得的方式分配服务名称，系统应用端口由工程任务组分配给一些标准的协议，如 FTP、POP3、SMTP。表 1-1 为一些典型端口及其所对应的应用服务。

随着 Internet 的不断发展，IANA 指定同意分配的端口不够分配，其他的应用使用某些公开的端口。随着这些变化，基于端口的网络流量分类技术，准确率逐渐减低，难以实现应用部署，现在一般只作为判断高带宽网络设备流量均衡的粗粒度基础。

表 1-1 常见端口协议映射表

端口号	应用服务名
20/21	ftp
22	SSh
23	telnet
25	SMTP
80	HTTP
110	POP3
443	https
53	DNS
67/68	DHCP
161/162	SNMP
3306	MYSQL
1433	SQLServer
6881	BitTorrent

1.2.2 基于 DPI 深度包检测

基于签名的分类技术使用深度包检测（DPI），这是一种监视技术，用于检查数据包的有效载荷以检测或分类特定类型的流量。对于 Internet 流量，这通常涉及到在包头之外的特定包位置查找值，或者扫描包内任意位置的字符串。

Sherry[13]提出 BlindBox，它是第一个同时提供这两种属性的系统。BlindBox 通过一种新的协议和新的加密方案来实现这种方法，演示了 BlindBox 支持 IDS、exfiltering detection 和 parent filter 等应用程序，并支持来自开源和工业 DPI 系统

的真实规则集。Sherry 实现了 BlindBox，并展示了它对于长时间使用 HTTPS 连接的设置是实用的，其核心加密方案比现有的相关加密方案速度加快 3-6 个数量级。Korczyński [14]提出了随机指纹，由于所选应用程序的指纹参数差异较大，该方法具有较好的应用程序识别精度，为检测 SSL/TLS 会话异常提供了可能。实验结果表明，获取应用程序其实主要来自不正确的实现实践、SSL/TLS 协议的误用、各种服务器配置以及应用程序的性质。

Hao[15]研究了细粒度的流量识别（FGTI），以便更好地理解和管理网络。而不仅仅指示哪个应用程序/协议的数据包与此相关，FGTI 将流量包映射到有意义的用户行为或应用程序上下文。首先提出规则组织最佳匹配（ROOM），分割标识规则分为几个领域，并精心组织匹配字段的顺序。因此，ROOM 只能激活对可能的小部分规则进行匹配操作。制定了最优规则组织问题 ROOM 在数学上证明它是 NP 难的，并且提出了一种启发式算法来解决问题 O（N^2）的时间复杂度（N 是规则集数据中的字段数）。基于 ROOM，进一步提出了 MP-ROOM 扩展到支持跨多个协议数据的规则用于流量识别的单元（PDU）。与此同时，实施了一个原型系统，包括 MP-ROOM 和相关的工作评估。实验结果显示：通过 MP-ROOM，系统获得了 71.3 倍的吞吐量改进，同时内存消耗低于 300 MB，并实现了真实系统多线程并行编程，成功实现了实际观测到的系统吞吐量超过 40 Gb / s。

Sen 等[16]通过运用深度包检测技术扫描应用程序签名来识别 P2P 流量。可以使用 Bro[17]、Snort[18]等入侵检测系统（IDS）软件进行深度包检测。例如，通过编写 Snort 规则，当字符串“torrent”运用 HTTP GET 命令获取的对象的名称出现时，该规则将提交使用 BitTorrent P2P 文件共享协议的主机。

在宽带 ISP 环境中使用的一些商业产品，如 Sandvine[19]和 ipoque[20]，使用专有的和硬件增强的技术，通过应用程序签名对流量进行分类。这种方法经常在非常大容量的链路上遇到性能障碍，并且很脆弱，例如，在面对分组有效载荷混淆时可能无效，已被证实这是一个持续存在的问题。此外，基于签名的分类方法在包抽样监控中效果有限。

一些商业产品使用经常包含域名的标识符执行流量分类和过滤。Websense 和 SmartFilter[21]等产品通过检查应用程序流量负载，可以确定标识符，如 url 等，还可以执行反向 DNS 查找。Alexa Internet 提供按域名标记的 web 流量指标，如顶级站点列表和人口统计数据。Alexa Internet 的服务针对特定 web 网址，并且观察统一资源定位器（url）。虽然 DNS 根据 IP 地址查询名称的能力可能非常有用，但是这种映射在当今的 Internet 中没有得到很好的维护。这种方法不可靠的主要原因是根本没有强制反向映射匹配其正向映射的机制，甚至不存在反向映射。这

种情况似乎还会继续存在，因为正向和反向映射的管理甚至被委托给不同的实体:一个是接收地址分配的实体，另一个是获得域名的实体。

1.2.3 基于协议解析

基于协议解析的流量分类方法通过分析特定应用的通信协议来描绘特定应用的特定行为特性。

Iliofotou[22]提出了流量散度图，使用 TDG 监测、分析和可视化网络流量。TDG 模拟了主机的社交行为，“谁与谁交谈”，边缘可以在哪里定义为表示不同的交互，例如交换一定数量或类型的数据包。随着 TDG 的引入，能够利用丰富的工具和来自不同学科的图形建模技术。

由于流行的 VoIP 应用 Skype 不仅采用私有通信协议，而且对通信数据进行了强加密，吸引了一些研究者深入探究其特性。Bonfiglio[23]分析了 Skype 客户端的特性，认为可将这些特性用于鉴别出入 Skype 客户端的语音流。他总结了 Skype 的一些流量特征并对一些真实呼叫进行了分析。但这个方法需要观察到完整的网络流，而且需要利用流中特定部分的数据块，因此只适用于边缘网络。Rossi[24]利用受控环境实验和被动测量分析 Skype 信令和流量特性。Bian[25]对 Skype 体系结构、网络路由、认证鉴别等问题进行了一些研究，分析了 Skype 的主要功能，如登录，NAT 和防火墙遍历，呼叫建立，媒体传输，编解码器和三种不同网络设置的会议。Suh[26]等人通过研究 Skype 中继结点特性，分析检测 Skype 中继流量。提出了指标描述 Skype 中继节点的特性，开始和结束时间差异，字节大小比，以及两者之间的最大的中继突发的数据包。

Constantinou[27]等人基于 P2P 协议的基本特性鉴别 P2P 流量。Subhabrata[28]等人对 P2P 协议及其签名进行了一些分析。Crotti[29]提出基于网络流估计的概率密度函数生成给定应用层协议的指纹并用于流量分类。Branch[30]提出了一种方法来探讨加密流量允许检测的 P2P 传输漏洞，创建了一个实验测试平台捕获多种流量，包括广泛使用的名为 GoalBit 的 P2P 媒体流应用程序。然后，分析收集的痕迹，并分析一组为 SNORT 网络入侵检测创建了规则系统，允许成功检测加密 GoalBit 生成的流量。这个系统的准确性通过了实验验证。Bermolen[31]提出了一种新方法来准确分类 P2P-TV 流量并识别生成它的特定 P2P-TV 应用程序。仅依赖于数据包的数量和在小时间窗口期间交换的字节，其基本原理是这两个计数传达了大量有用的信息，涉及应用的几个方面及其内部工作原理，如信号活动和视频块大小。分类框架使用支持向量机准确识别 P2P-TV 流量以及由其他类型的应用程序生成的流量，以便误分类事件的数量可以忽略不计。通过使用测试平台和实际网络流量的大型实验活动，证明它通过简单地计数分组来分类不同地区的 P2P-TV 应用实际上是可行的。Zhou[32]对 P2P 流量的 TCP 特征进行了一些分析；

Dhamanka[33]提出一种鉴别加密协议或匿名使用已知端口的应用的框架，基于对数据包和网络流的统计分析，将每种应用协议映射到十维空间，以便使用 SSH 和 Skype 进行群集和测试。

Maciej[34]提出随机指纹用于在安全套接字中传送的应用程序流量层/传输层安全性（SSL/TLS）会话。指纹基于一阶齐次马尔可夫链，其中从观察到的训练中识别出参数应用痕迹。作为选择的指纹参数应用差异很大，该方法识别应用的效果非常好，并提供了检测异常 SSL/TLS 会话的可能性。

Yang[35]将无源网络监控设备放置在互联网骨干，收集了海量的网络数据，深入提供了当前的中国当前互联网的 P2P 流媒体流量。特别是两个 P2P 流媒体流量，有线（ADSL）和无线（CDMA）网络的统计研究，并描述来自流级别和数据包级别的流量方面。结果揭示了 P2P 的有线和无线网络中的流媒体流量由于它们各自的内在环境特征，导致特征之间的显著差异。

Maciej[36]检测了 Skype 流量，并将 Skype 流量分类为语音通话，SkypeOut，视频会议，聊天，文件上传和下载。提出了基于统计协议标识（SPID）的 Skype 加密流量的分类方法，通过分析某些流量属性的统计值来检测 Skype 流量。

当用户使用由 Secure Shell（SSH）提供的加密隧道时来保护他们的隐私时，用户期望两种形式的保护。首先，是保护其数据的隐私。其次，用户期望他们使用的应用程序的类型也是保持私密的。Dusi[37]提出了两个可用于打破第二种类型的分析技术应用于 SSH 隧道时的保护，至少在某些情况下限制假设。实验结果显示当前 SSH 的实现很容易受到这种类型的分析的影响，并说明了提出的两个分类器在分析加密流量和相对计算复杂性的有效性。

1.2.4 基于统计学习

基于统计学习的流量分类方法通过计算网络流量的统计特征和使用机器学习算法，对网络流量分类。

Zeng[38]提出了一个轻量级框架，借助深度学习来对加密流量分类和入侵检测，称为深度全范围（DFR）。通过深入学习，DFR 能够从原始流量中学习，无需人工干预和私密的信息。将 DFR 框架与当前其他最先进的方法在两个公共数据集的进行了比较。实验结果表明，DFR 框架不仅可以通过在加密流量分类的 F1 分数上平均为 13.49%，在入侵检测的 F1 分数上平均为 12.15%，而且只需要更少的存储资源需求，相比当前其他最先进的方法。

Anderson[39]分析了六种常见的机器学习算法，并展示了它们各自在检测恶意加密网络会话问题上的表现。发现随机森林集成分类器对于这个问题来说是最健壮的，证明了特征的选择对性能有更大的影响。增强的特性集是通过将该域中使用的标准特性集与域专家标识的特性进行扩充而创建的，这些特性集是为加密的

网络会话专门定制的。所有的机器学习算法并不仅仅依赖于便于收集的特性，也不需要与领域专家反复讨论如何最好地表示数据，因此它们的性能都有了显著的提高。

Saber[40]使用新的加密流量分类模型来自动识别互联网上的活动，不仅为网络管理提供基本的技术支持，还可以增强网络安全性。所提出的方法 LDA-KNN 结合了线性判别分析（LDA）特征提取和降维，与 K-最近邻（KNN）算法相比能更好地区分各个类别。结果表明，LDA-KNN 比以前的分类器具有更好的性能，证明了 LDA-KNN 的有效性。

在基于机器学习的移动应用流量分类中，由于网络环境的变化，用户习惯等，流量特征分布很容易漂移。不稳定的特征可能会对移动应用流量分类的稳健性产生负面影响，因此为了解决这个问题，Liu 等[41]研究了如何获取用于提高移动应用流量分类稳健性的最佳功能集。具体而言，通过联合分析移动应用流量特征和评估特征漂移程度，开发了一种搜索稳定和判别特征的方法。首先分析流量的流入行为特征，以便为移动应用流量数据提取潜在的特征集。接下来，提出两个新的指标来评估不同视角下流动特征所经历的漂移程度，并设计了一个综合指标，通过将漂移程度视为辨别力的惩罚因子来对这些特征进行评分。基于这些指标，进一步提出了一种算法来搜索具有高辨别力但低漂移度的最佳特征。为比较性能，实验实施了现有的流动特征和特征选择算法，在真实移动应用流量数据上的实验结果，证明了特征集和特征选择算法在提高分类鲁棒性方面的有效性。

Shekhawat 等[42]将三种机器学习技术应用于区分恶意加密 HTTP 流量与良性加密流量的问题，获得与之前工作相当的结果，并详细考虑了特征分析的问题。以前的工作通常依赖于人工专业知识来确定此问题域中最有用和最有用的功能，现在这些与特征相关的信息可以直接从机器学习模型本身获得。这种基于机器学习的特征分析方法更可取，因为它更可靠，例如，可以发现特征之间相对不直观的交互。

类别间的不平衡已经成为一个导致网络流量分类不准确的大问题。准确的流量分类有助于进行安全监控，IP 管理，入侵检测等。Shafiq 等[9]提出了一种基于机器学习的混合特征选择算法 WMI_AUC。该算法利用两个度量：加权互信息（WMI）度量和 ROC 曲线下面积（AUC），这些指标从流量中选择有效功能。但是，为了从所选特征中选择鲁棒特征，提出了鲁棒特征选择算法。所提出的方法提高了机器学习分类器的准确性并有助于检测恶意流量。实验使用 11 个著名的机器学习分类器在不同的网络环境跟踪数据集上评估，实验结果表明，算法实现了 95％以上的流量精度结果。

Sun[11]首先介绍了增量 SVM（ISVM）模型，以降低内存和 CPU 的高训练成本，实现了流量分类器的高频快速更新。带有衰减因子的 ISVM 模型的修改版本称为 AISVM，进一步提出利用先前训练数据集中的有价值信息。实验结果证明了 ISVM 和 AISVM 模型在流量分类中的有效性。

Niu[43]提出了一种启发式统计测试（HST）方法，它结合了统计学和机器学习，并已被证明可以改善各自的不足。文中手动选择了四个随机性测试来提取小型有效载荷功能以进行机器学习，从而提高实时性能。同时还提出了一种名为 HST-R 的简单握手跳过方法，以提高分类精度。文中与其他识别方法进行了比较，测试数据集包含使用两个已知，两个未公开和一个自定义加密协议的流量。实验结果表明，HST-R 比其他传统的基于编码、基于熵和基于机器学习的方法表现更好。实验结果还表明，简单握手跳过方法可以更好地适用于未知的加密协议。最后，在不同的分类算法之间进行了对比实验，结果表明，C4.5 和 HST 对安全套接层和 secure shell 流量具有最高的识别精度。

Sun 等[44]提出了一种转移学习模型。最大熵模型（Maxent）用作转移学习模型中的基本分类器。为了测试所提方法的有效性，在剑桥大学收集的流量数据集种训练和测试不相同的数据集。实验结果表明，基于转移学习模型可以获得良好的分类性能

Dong 等[45]提出了一种新的应用程序识别方法 MPNN，以提高应用程序识别的效率和灵活性。MPNN 使用基于多个神经网络的结构，该结构使用单独的神经网络模块来处理单个应用程序。因此，它可以有效地利用每个应用的特点。同时，在每个神经网络模块中使用最小贝叶斯方法。MPNN 方法具有以下优点：它可以处理更加复杂多变的网络行为，识别的对象也可以从完整的 TCP 流扩展，直到识别所有 TCP + UDP 流。它可以提高每个应用程序的识别准确性，特别是对于包含比其他应用程序少得多的流量的应用程序，更改已识别的应用程序的过程变得更加容易。由于采用并行处理，它具有更低的时间和空间复杂度。理论分析和实验结果表明，MPNN 可以达到 95％的识别精度。

Jie[46]提出了一种实时精确的 SVM 训练模型 SPP-SVM。从缩放数据集中扣除 SPP-SVM，并使用主成分分析（PCA）来提取数据特征并验证从 PCA 获得的相关流量特征。通过采用 PCA 算法进行尺寸提取，SPP-SVM 确定了关键部件的特征，减少了它们之间的冗余，降低了原有的特征尺寸，从而有效地减少了过拟合，增加了其泛化。利用改进的粒子群优化算法自动导出 SPP-SVM 中核函数的最优工作参数，优化全局解，使其惯性权重系数自适应，而无需在大范围内搜索参数，遍历所有参数点在网格中逐步调整步骤。它的两级和多级分类器的性能通过互联网上不同拓扑点的两组流量轨迹得到证明。实验表明，SPP-SVM 的两类

和多类分类器优于典型的监督 ML 算法，并且在分类精度、维度和计算时间方面表现明显优于传统 SVM。

Piskac[47]提出了使用有关数据包大小和网络流的时序特征统计信息，来检测网络流量类别。与此同时，通过使用聚合的 IPFIX 数据而不是查看数据包内容来处理 DPI 在计算开销方面的劣势。此外，应用程序独特的行为模式，也可用于应用程序检测。

Wang[48]首先从人工智能的角度提出了一种新的流量分类，接着提出了一种以流量数据为图像，基于卷积神经网络的恶意软件流量分类方法。该方法不需要人工挑选特征，可以直接将原始流量作为分类器的输入数据。这一有趣的尝试是第一次将表示学习方法应用于使用原始流量数据的恶意软件流量分类。通过八个实验确定了最佳流量表示方法。实验结果显示，该方法能够满足实际应用的精度要求。

Mohammad[5]提出了基于深度学习的 Deep Packet 方法，将特征提取和分类阶段集成到一个系统中。与大多数现有方法不同的是，Deep Packet 不仅可以识别加密流量，还可以区分 VPN 和非 VPN 网络流量。

Shi[6]提出了新颖的特征提取和选择方法。首先，使用 Wavelet Leaders Multifractal Formalism 从网络流中提取多重特征来描述网络流量。接下来，使用基于 FS 方法的主成分分析（PCA）筛选特征，以消除不相关和冗余的特征。实验结果显示，与现有的基于机器学习的方法研究的 TLS 特征相比，SVMs 的准确性有显著提高。此外，由于能够在网络传输的早期阶段对网络流量进行分类，所提出的方法也适用于实时的网络流量分类。

Yang[7]提出了适用于网络视频细粒度流量分类的特征，并从真实网络中收集数据。这些特征是与体验质量（QoE）相关的参数，其反映了用户的感知。QoE 值基于 ITU-T P.1201 / Amd2 标准计算。在该标准下，每个视频流可以计算相应的 QoE 值及其分布概率，提取了 QoE 值的特征及其概率分布作为适用于视频流量分类的区分特征。提取的 QoE 分布特征通常是均值、方差、最大和最小统计特征，并且可以获得特征的概率分布。首次获得了五个值的概率离散分布，并将它们直接用作独立的特征来参与特征选择和分类。实验结果显示，与现有方法相比，所提出的新特征可以显著提高分类精度。

Shen[49]首先探索一些可以进一步提高现有的在识别准确性方面的性能的方法，并观察到应用程序属性 bigram（由 SSL/TLS 会话中的证书包长度和第一个应用程序数据大小组成）有助于应用程序识别。为了增加应用指纹的多样性，提出了一种将属性二图引入二阶齐次马尔可夫链的新方法。仿真结果表明，与最先进的基于马尔科夫的方法相比，该方法能将分类准确率平均提高 29%。

Zuleika[50]提出了一种混合模型，该模型利用基于计算智能的分类器、极端学习机（ELM）、特征选择（FS）和多目标遗传算法（MOGA）对计算机网络流量进行分类，而不需要利用负载或端口信息。在使用四种性能指标对 UNIBS 数据集评价时，所提出的模型显示了良好的结果，其中大多数超过 90%。此外，给出了给定问题的最佳特征和特征选择算法以及最佳 ELM 参数。

Yang[21]提出了一种新的前 N 个分组滑动窗口的加密网络流量分类算法，该方法可以明显降低流量特征特征维数，减少每个流量中的数据包数量。Wang[51]提出了一种基于一维卷积神经网络的端到端加密流量分类方法。该方法将特征提取、特征选择和分类器集成到统一的端到端框架中，自动学习原始输入和期望输出之间的非线性关系。并通过公共 ISCX Vpn-Non- Vpn 流量数据集对分类方法进行了验证。在 4 个实验中，通过对流量的最优表示和模型的微调，实验结果的 12 个评价指标中有 11 个指标优于目前的方法，说明了所提方法的有效性。Giuseppe[52]将深度学习作为一种可行的策略，设计了基于自动提取特征的流量分类器，以反映复杂的移动流量模式。并复制、剖析了来自 TC 的不同技术状态的深度学习技术，并将其设置为一个系统的比较框架，其中包括一个性能评估工作台。基于三个真实的人类用户活动数据集，对这些深度学习分类器的性能进行了严格的研究，突出了移动加密 TC 中深度学习分类器的缺陷、设计准则和开放问题。Pan[53]针对最近提出的几种侧重于在各种 SSL/TLS 应用程序之间识别定义指纹的方法显示出的各种局限性，设计了一个加权集成分类器（WENC）来解决这些限制。WENC 研究 HTTPS 握手过程中各个子流的特征以及接下来的数据传输周期。为了提高指纹识别率，提出了一种基于指纹变量的二阶马尔可夫链模型，该模型综合考虑了 HTTPS 握手过程中的数据包长度和消息类型。最后，设计了一种加权集成策略，以适应多种方法作为统一方法的优点。决策树的代表方法有 ID3[54]、C4.5[55]等。支持向量机（support vector machine，SVM）是 Corinna Cortes[56]等人于 1995 年首先提出的。Xiang[57]提出半监督支持向量机方法仅基于网络流统计特征来识别并对网络应用进行分类。在这种方法中，SVM、恒定流和共同训练算法是快速获得分类器的关键。通过这种方法得到的分类器有三个优势，与以往的经典方法形成对比：高分类度、高泛化性能、快速的计算性能。Bar[58]提出了一种新的统计分类器，将基于 K 均值和最近邻（或 k-NN）几何分类器混合组合，对加密数据进行实时分类。Tan[59]介绍了应用统计模式识别方法，最大似然分类对 SSH 隧道流量进行分类。Andres[60]提出了基于特定域的每跳行为（PHB）的 VPN 流量的特定 QoS 分类器。从表征的 VPN 流量生成基线 QoS 标记数据集，比较了一些机器学习算法并执行了 T-Tester。其中，基于 Bagging 的学习模型在所有场景中具有最佳行为，所获得的准确度的较高值是 94.42％的。因此 QoS 分

类器是差分服务（DiffServ）网络上的流量处理的有效方法。Gerard[61]研究了运用基于网络流的时间相关特征检测 VPN 流量的有效性，并根据流量类型，如 Email，将加密流量表征为不同类别。使用两种不同的机器学习技术，C4.5 和 KNN，来测试分类能力，结果显示出高精度和高性能，确认了与时间相关的统计特征是加密流量表征的良好分类器。Meng[62]提出了一种基于二阶马尔可夫链证书感知加密流量的分类方法。探讨了现有分类方法的缺点，发现认证在 SSL / TLS 会话的数据包长度有助于识别应用。为了增加应用特征的多样性，将认证数据包长度聚类成二阶齐次马尔可夫链。通过广泛的评估，结果表明，该方法与最近方法相比较，平均提高了 30％分类准确率。Fu[63]开发了一个名为 CUMMA 的系统，用于联合分类移动消息应用的服务，使用建模用户行为模式，网络流量特征和时间依赖性。沿着这条线，首先划分互联网来自流量的流量，以分层方式流入具有多个对话的会话。此外，提取流量的特征来自两个方面的数据，分组长度和时间延迟。接下来，学习服务使用预测器来对这些分段进行分类对话框分为单一类型的用法或异常值。此外，还设计了一种基于聚类隐马尔可夫模型（HMM）的方法来检测混合来自外部的对话框，将混合对话框分解为单一类型用法的子对话框。事实上，CUMMA 使移动分析师能够做到，即使是加密的互联网流量，也能识别服务使用情况，并分析最终用户的应用内行为。最后，进行了广泛的实验，证明了所提出的服务使用分类方法的有效性和有效性。

传统上，用于流量分类的签名是在字节级构造。但是，越来越多网络协议和应用程序的数据传输格式在比特级编码，字节级签名正在丢失它们在交通分类中的有效性。Yuan[64]通过关联来创造性地构建位级签名在每个业务流中的位置。此外，还提出自动化流量挖掘 BitMiner 可以挖掘应用程序签名的工具细粒度的比特级粒度。初步测试流行的点对点（P2P）应用，例如 Skype、谷歌环聊、PPTV、eMule、迅雷和 QQ 下载，显示了尽管它们都没有字节级签名，但流量中依然隐藏着的重要的位级签名。Sun[65]提出了一种混合方法，结合基于签名和统计分析的方法对加密流量分类。首先确定 SSL / TLS 流量用签名匹配方法，然后应用统计分析以确定具体的应用协议。实验结果表明，该方法能够识别超过 99％的 SSL / TSL 流量，并达到了 94.52％的 F-score。Bernaille[66]提出了一种检测应用程序的方法 SSL 加密连接。仅通过检测前几个数据包的大小就可以用于识别应用程序的 SSL 连接，可用于实现早期的网络流量分类。实验测试了在两个校园网络上收集的数据包，结果表明能够识别到 SSL 连接中的应用程序，准确率超过 85％。Bonfiglio[67]提出一个基于两种互补技术的框架实时识别 Skype 流量。第一种方法基于 Pearson’s 卡方验证，并且与 VoIP 相关的流量特性无关，用于从数据包帧中检测 Skype 的指纹结构，利用在位级引入的随机性加密过程。相反，第二种方法

是基于 Skype 流量的数据包的随机特征到达率和包长度，用作决策的特征，并采用朴素贝叶斯分类器。为了评估上述技术的有效性，开发了基于深度包的离线交叉检查启发式算法。从不同网络流量的测量结果显示该技术在识别 Skype 流量方面非常有效。Erman[10]运用两个无人监督聚类算法，即 K-Means 和 DBSCAN，评估这两种算法并与之前使用的 AutoClass 算法进行比较。实验结果表明，K-Means 和 DBSCAN 都非常有效，比 AutoClass 的计算速度快。尽管 DBSCAN 与 K-Means 和 AutoClass 相比准确度较低，但 DBSCAN 能产生更好的集群。

Bernaille [68]发现可用 TCP 连接中的前五个数据包来识别应用程序。Moore[69]使用朴素贝叶斯按照应用程序对网络流量分类。

能源是无线计算系统的重要资源。尽管无线局域网（WLAN）日益普及，但最重要的突出问题之一仍然是无线网络接口控制器造成的功耗。为了节省这种能量并降低无线设备的总功耗，迄今为止提出的大多数方法都集中在静态和自适应省电模式上。现有的研究强调了其功耗和性能下降方面的若干问题和限制，需要进一步增强。Saeed[70]提出了一种新的基于机器学习（ML）分类器的上下文感知网络流量分类方法，用于优化 WLAN 功率节省。分类输出流量用于优化提出的上下文感知监听间隔省电模式。基于九个智能手机应用程序的网络流量记录了反映不同类型的网络行为和交互的真实数据集。这用于评估八种 ML 分类器在该初始研究中的性能。比较结果表明，可以实现 99％以上的精度。ML 分类器适用于根据后台交互级别对智能手机应用程序的网络流量进行分类。Giuseppe[71]提出深度学习作为一种可行的策略，即基于自动提取的功能设计实用的移动流量分类器，能够处理加密的流量，并反映其复杂的流量模式。为此，来自流量分类的不同最先进的深度学习技术在这里被复制，并且设置成用于比较的系统框架，包括性能评估工作台。后一种结果虽然在移动环境中有所下降，但其适用性对更广泛的加密流量分类任务具有吸引力。最后，基于详尽的实验验证，基于真实人类用户活动的三个移动数据集，重点研究了这些深度学习分类器的性能。

互联网网络的内在特征导致数据集的类分布不平衡一致的，称为不均衡类别，在许多研究领域已经引起人们注意。尽管由于类别失衡导致性能下降，在网络流量分类领域但这个问题还没有彻底研究过，以前的一些工作仅限于少数解决方案或假设误导性的方法论方法。Santiago[72]整理了网络流量分类中不均衡类别问题，分析并研究了这种现象的存在，并讨论了两种不同互联网环境中的大量解决方案：实验室网络和高速骨干网。同时，尝试了 21 种数据级算法，6 种集成方法和 1 种代价级方法。在整个实验过程中，将最新的方法应用于数据集不平衡问题，例如：DOB-SCV 验证方法或假设的性能指标。调整参数的策略和算法不仅适用于二分类也适用于多分类问题，包括两个机器学习中首次使用的技术。实验结果显示，

一些技术减轻了类不平衡，为流量分类模型带来了有趣的好处。更具体地说，一些算法的总体准确度增加了超过 8%，在最具挑战性的网络方案中，AUC-ROC 大于 4%。Lim [73]通过网络流量预处理生成了基于数据包的数据集，运用卷积神经网络（CNN）和剩余网络（ResNet）训练五个深度学习模型来执行网络流量分类。最后，使用 CNN 和 ResNet 深度学习模型的 F1-Score 分析了基于分组的数据集的网络流量分类性能，并证明了其有效性。Sun[74]提出了一种多输出 DNN 模型，同时学习多任务流量分类。在该模型中，通过任务之间的协同作用并利用流量的常见的特征，分别提高每个任务的性能。结果表明，该结构具有满足未来新需求的潜力，同时能够以很快的速度和公平的准确度实现分类。还探讨了 one-shot 学习，即学习稀缺数据的学习过程，也表现出优越的性能。

随着通信计数、网络计数不断发展，互联网新计数、新应用层出不穷，用户规模日益扩大，用户的使用模式和行为持续演进，网络传输带宽呈爆炸性增长。网络流量的快速增长要求骨干网不断优化演进。这一切使得流量特性不断发生变化，从而使得本未完善解决的网络流量分类问题再次面临巨大挑战。

第一是高宽宽带来的数据实时无损采集的挑战，当前，网络带宽的增长速度远高于微处理器集成度和性能的增长速度。根据贝尔定律可知芯片集成度会在约 18 个月增加一倍。根据摩尔定律可知处理器的处理速度每 18 个月会增加一倍。根据吉尔德定律：骨干网带宽每 6 个月增长一倍。但总体来看，网络带宽的增长速度远高于微处理器集成度和性能的增长速度。当前边缘网络出入口的带宽普遍在 1Gbit/s 到 10Gbit/s 之间，单点聚合带宽可以达到几百 Gbit/s。软件层面通过修改操作系统内核，采用零拷贝，综合满足高带宽链路的数据包捕获需求。

第二是网络扁平化、流量均衡导致的单向流及系统时钟精度对现有分类算法的挑战。大量的网络采用动态路由算法，网络流量在多条链路或多个异地出入口之间动态调配，导致在流量观察点仅能看到部分流量，其中一些流只有单个方向的数据包。另一方面，当前多数基于统计学习分类方法的研究成果普遍表明包到达间隔时间是分类时的一个极佳的区分特征，但由于网络速度大幅提升，数据包之间的到达间隔因系统时钟精度等原因变得不明显，从而需要进一步优化调整算法。

第三，与互联网早期倡导的完全开放精神相背离，当前互联网已呈现出基于通用 TCP/IP 协议承载私有应用的趋势，各种应用如雨后春笋般涌现。至 2019 年底，苹果和安卓的应用均超百万。这些应用均采用各自的应用层协议，与主流应用相比既难以识别又难以细分。此外，一些主流应用如 Skype、VoIP 应用及大多数网络聊天工具均采用私有加密协议及加密通信的应用程序，没有公开标准规范

可以解析得到其协议的准确行为特征行为和签名，甚至无法通过实验方法枚举这些应用的行为以得到可用的样本流量。

第四是 P2P 传输全面普及带来的传输和应用分离、传输和通信断电分离给采集有效数据带来的挑战。P2P 结点在行为下兼具资源提供者和资源使用者双重角色，而 P2P 网络相当于在 TCP/IP 上又增加了一层网络承载和传输协议，不仅完全扰乱了传统 C/S 或 B/S 访问模式下通信端点之间的一一映射关系，在网络上的某个观察点仅能观察到部分流量，而且由于 P2P 的流量模型与传统 TCP/IP 网络承载的传统流量模型有显著差异，使利用传统流量模型构造的流量分类算法不再适用。

第五是协议混淆、云平台给应用细分带来的挑战。一些应用为逃避检测，往往冒用一些已知应用的默认端口，仿冒已知协议，如 HTTP、SHTTP、SMTP 等协议的通信规程和数据包格式，提高了流量分类算法的误检率。另一方面，越来越多的应用程序部署在第三方公有云平台上。云平台通过一个动态伸缩的架构，提供了一个庞大的计算和存储资源池，对外提供应用程序在云上的部署和运行服务。由于云平台对应的动态部署特征，一群服务器上往往承载了一组不同应用，出入云平台的网络流量混合了多种不同应用的流量，屏蔽了单个应用的流量特征。另外，由于云平台对应用部署经常动态调整，更难以获得云流量的可信训练样本，对云流量进行识别和类别细分极为困难。

第六是普遍采用加密和隧道化承载进行通信给载荷分析带来的挑战。从隐私保护考虑，除越来越多的应用自身增加加密传输机制外，一些加密隧道也已普遍被用来承载非加密应用，可以为各种应用提供适用的加密传输隧道。流量分类方法不仅要考虑检测出这些加密和隧道流量，而且到进一步识别出其承载的应用则显得更为困难。

最后，在网络上不同的观察点（如核心骨干网、边缘网络及主机接口处）所看到的流量具有不同的统计特性。流量分类算法不仅要求具有时间稳定性，即能以低载进行及时连续分类，又要具有空间适应性，即能适应网络架构及流量特性的变化，具备移植的鲁棒性和方向中立性，支持基于单向流或双向流进行分类，而这些要求往往是多数统计学习分类方法的致命伤。

限制网络流量分类发展的最大问题还有包括互联网网络隐私保护带来的法律支持，缺乏基准数据集，导致不同研究之间无法开展对比性评估，使面向小规模网络的全流量开展的研究无法适用于大规模分布式网络。

1.3 本文主要工作

当前，基于统计学习的机器学习方法被应用于流量分类，通常使用经典的机器学习方法，例如朴素贝叶斯和支持向量机，然后这些算法的分类能力依赖于人

工对于特征的挑选，无法广泛应用。而深度神经网络能在训练中自动选择特征，而不需人为提前挑选。当新的网络流量类型出现或是已有的流量特性发生改变时，深度神经网络依然能很好的对流量分类。深度神经网络相比传统的机器学习方法，学习复杂情况的能力更强，能更好学习原始输入与输出之间非线性的关系。因而，深度学习更适合用来进行加密流量分类研究。

本文主要工作如下：

1．分析现有的网络流量分类方法。

2．运用 ISCXFlowMeter[61]生成网络流，并提取经由 VPN 加密的网络流基于时序的特征。

3．设计并构建了基于深度神经网络的加密流量分类模型，对多种经由 VPN 加密的 HTTP、Email、P2P、Chat、Browsing、FT、VoIP 等 7 种网络流量进行分类，并分析影响分类模型对不同类型网络流量分类的能力的原因。

4．在深度神经网络中，影响分类模型分类能力的因素众多，本文主要探讨了不同 mini-batch size 对基于深度神经网的加密流量分类模型分类能力的影响。在训练时，mini-batch size 会对深度神经网络模型产生不同程度的影响，当 mini-batch size 为 40 时，深度神经网络模型的分类能力最好。

5．本文将基于深度神经网络的加密流量分类模型与朴素贝叶斯分类算法对比，探究不同的分类算法对加密网络流量的分类能力。实验结果表明，基于深度神经网络的加密流量分类模型能够很好的识别并分类加密流量。

6．在本文采用的网络流量数据集中，各个类别的网络流量样本数量差异明显，本文也研究了多类样本数量不均衡数据集对分类的影响。

1.4 论文的组织结构

本文共有六章，主要结构如下：

第一章为绪论，主要分析了网络流量分类研究的必要性，介绍了网络流量分类领域的历史与现状，并概述了本文主要研究内容与论文的组织结构。

第二章主要分析了网络流量分类问题，简单概述了基于统计学习的网络流量分类过程。分析了深度学习中的深度神经网络算法，以及朴素贝叶斯算法，最后总结了网络流量分类中常用的统计特征。

第三章首先介绍了经由 VPN 传输的网络流量使用的 SSL/TLS 协议的工作流程。然后，介绍了采集加密网络流量的方法以及捕获的数据包的文件格式。描述了本文使用的数据集的组成，以及数据集的预处理方式。

第四章主要介绍了基于深度神经网络的加密流量分类模型整体架构，主要包括加密网络流量的数据集采集、数据集预处理过程。基于深度神经网络的加

密流量分类模型算法分析，训练过程，分类模型的时间复杂度分析，和 mini-batch 下降法的原理。

第五章，实现了基于深度神经网络的加密流量分类模型，并运用基于深度神经网络的加密流量分类模型对 7 种加密网络流量分类。同时运用朴素贝叶斯算法对加密网络流量分类。随后，对比两种加密流量分类算法。

第六章，总结与展望。总结本文的主要工作，并提出未来研究的内容。

第 2 章 网络流量分类相关技术概述

本章首先介绍了网络流量分类问题的定义，描述了基于统计学习的网络流量分类的一般过程。然后详细介绍了用于网络流量分类中的两种机器学习算法，深度神经网络和朴素贝叶斯算法。最后，介绍了网络流量分类的统计特征。

2.1 网络流量分类的定义

网络流量分类是指网络流量使用的协议或者网络流量的类型，根据其各自的特点，将网络流量分成不同类别的过程。

骨干网或核心网处于网络可靠性需要，一般会采用分布式非对称路由，因此引出单向流问题，即同一个网络流的出方向数据包和入方向数据包会经由不同的网络链路。在观察点只能看到单个方向的数据包，单个方向的流被称为单向流，与之相对，完整的网络流被称为双向流。

2.2 基于统计学习的网络流量分类过程

基于统计学习的网络流量分类方法首先提取一组网络流量统计特征，然后利用机器学习算法训练网络流量分类模型并调整分类模型的参数，然后将分类模型用于网络流量分类。

2.3流量分类方法

2.3.1 深度神经网络

深度神经网络是由输入层、隐藏层和输出层组成的多层神经网络。深度神经网络能在输入与输出之间找到正确的数学变换，通过网络层来计算每个输出的概率。图 2-1 为深度神经网络分类模型。

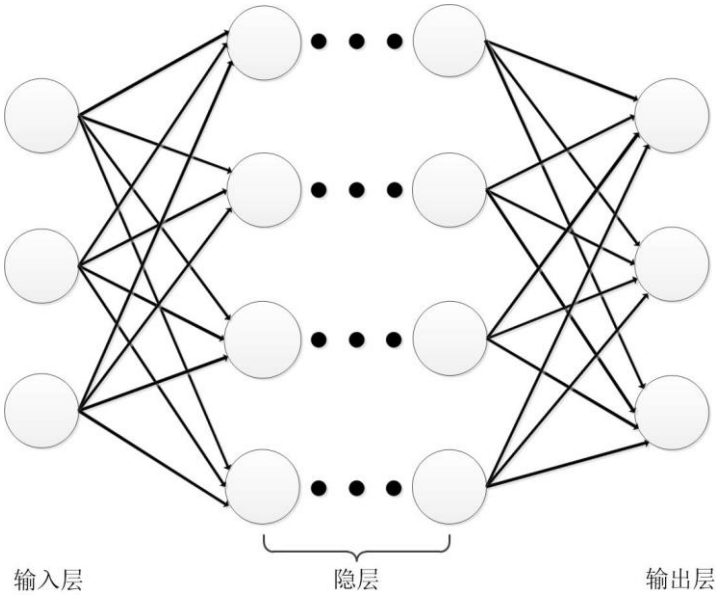

图 2-1 深度神经网络分类模型

深度神经网络每一层由多个神经元构成，每个神经元都是具有多个输入和多个输出的模型。

在过去的几年中，深度神经网络（DNN）已经成为一种强大的机器学习分类模型。DNN 与传统的分类方法存在重大差异，深度神经网络是深层架构，可以学习比浅层神经网络更复杂的模型。深度神经网络这种表现力强大的训练算法能够学习强大的对象表示，而无需手工设计功能。通过对不同深度学习模型的研究，得知具有最佳检测性能的模型是 CNN 和 RNN 的组合。每个 CNN 层生成多维数组（张量），其中图像的尺寸减小，但同时生成新的尺寸，使该新尺寸等于应用于图像的尺寸的数量。连续的 CNN 层将进一步减小图像尺寸并增加新生成的尺寸。当连接多个 LSTM（Long Short-Term Memory）层时，所有 LSTM 层（除了最后一个）采用返回序列模式，该模式产生对应于循环网络的连续迭代的向量序列。该矢量序列可以按时间顺序分组，形成下一个 LSTM 层的入口点。对于连续的 LSTM 层，数据输入的时间维度不会改变，但是连续输入的矢量维度会改变。

此外，还可以使用一些额外的层：批量标准化，最大池化和丢失层。批量标准化使训练收敛更快，并可以改善性能结果。这是通过在训练时将批量级别的每个特征标准化（将输入缩放到零均值和单位方差）并在以后考虑整个训练数据集再次重新缩放来完成的。新学习的均值和方差取代了批量级别获得的均值和方差。最大池化是一种卷积层，不同之处在于使用过滤器。在最大池化中，它使用最大值并选择应用过滤器的图像区域的最大值。它减少了输出的空间大小，减少了特征的数量和网络的计算复杂性。结果是下采样输出。与丢失层类似，最大池化层提供正则化。丢失层则是通过从前一层中丢弃（设置为零）一定百分比的输出来提供正则化（对看不见的数据的结果的推广）。这种明显无意义的行为迫使网络不要过度依赖任何特定的输入，从而超越和改进泛化。

深度神经网络的大体训练步骤一般为：

1．定义神经网络

2．定义 loss，选择优化算法，使得 loss 最小

3．迭代训练数据，使 loss 最小

4．在测试集上计算分类模型的准确率

2.3.1.1 神经元模型

计算生物神经元模型和神经网络的创建是一个不断发展的研究领域。计算模型是验证和测试对生物过程（包括生物神经网络）的理解的重要工具。由于 ModelDB 和开源脑库（Open Source Brain）等模型库的努力，计算神经科学中的模型变得越来越容易获得。大多数神经元和神经网络模型以模拟器特定格式实现，或者作为 C 或 Java 等编程语言中的定制专用模拟器的一部分实现。如果没有特

定的模拟器设置或每个软件包的正确版本，这可能使重现计算模拟变得困难。这种困难突出了对可重现和易于理解的建模规范的需求，并促进了 NeuroML 建模语言的发展。NeuroML 语言包含用于构建神经元模型和网络的特定构建块。NeuroML 依赖于每个兼容的神经模拟器，实现标准模型的一小组标准组件。然而，由于一些模拟器以不同方式实现基本神经构建块。因此，NeuroML 定义的模型在任何模拟器中仍然不能完全重现。

如图 2-2 是典型的神经元模型，其中神经元包含x_1, x_2, x_3三个输入和一个输出 output，输入数据x_1, x_2, x_3由神经元的计算，输出信号的结果 output，每个神经元都有权值 w，以及激活函数 g，神经元的输出 output 是：

$$\text{output} = \text{g}(w_1 * x_1 + w_2 * x_2 + w_3 * x_3) \tag{2.1}$$

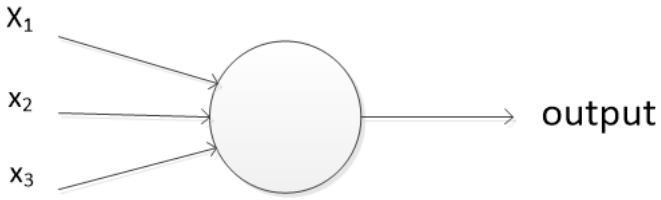

图 2-2 神经元模型

2.3.1.2 感知机

感知机由两层神经元组成，如图 2-3 所示，x_1和x_2接受外界输入信号后传递给 y。

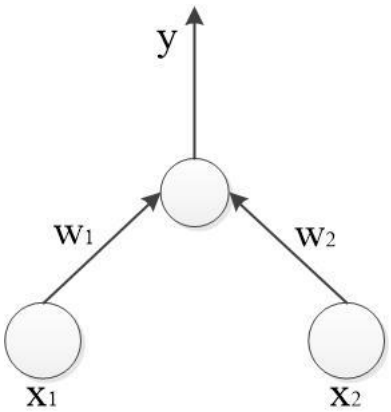

图 2-3 感知机

2.3.1.3 线性整流函数 ReLU

激活函数决定节点根据输入生成怎样的输出。ReLU 是一种经典的激活函数，其有效性已在以前的工作中得到验证。ReLU 的成功主要在于传输所有相同地正输入，这缓解了梯度消失，并允许对更深层次的神经网络进行监督训练。此外，ReLU 通过仅为负输入输出零来提升计算效率。 因此，它广泛用于神经网络。尽管 ReLU 很棒，但研究人员发现激活功能不是研究的终点。

ReLU 的以下三种变体利用了已经压缩的负面部分信息。负部分的斜率始终在（0，1）的范围内。ReLU 只保留输入的正面部分并将负面部分修剪为零。主要试图改变 ReLU 以不同方式处理负输入。Leaky ReLU 为负部分指定非零斜率，以避免零梯度。也就是说，负部分的斜率是预定义的。然而，实验结果表明，与 ReLU 相比，LReLU 对精度的影响可以忽略不计。参数化 ReLU（PReLU），其中负部分的斜率是在训练期间从数据中学习而不是预定义的。PReLU 可以通过端到端的训练，与整个网络模型一起自适应地学习斜坡的参数。无论是采用预定斜率还是学习斜率，LReLU 和 PReLU 都利用具有恒定斜率的线性函数来表示负部分。随机泄漏整流线性单元（RReLU）不是使用恒定斜率，而是在训练期间随机采样负部分的斜率，这有助于降低过度拟合的风险。

激活功能的挑战来自两个主要方面：负面缺失和零样性质。负面缺失：ReLU 只是将负值限制为零，这提供了稀疏性，但导致负面缺失。ReLU 的变体，包括泄漏 ReLU（LReLU），参数 ReLU（PReLU）和随机化 ReLU（RReLU）使负极部分变成非零斜率。这些方法表明负极部分有助于神经网络的构建。然而，他们不稳定的整改会破坏稀疏性。零样性质：LReLU，PReLU 和 RReLU 无法确保噪声稳健的负失效状态。为此，提出了指数线性单位（ELU）来保持负值并使负极部分饱和以实现类零特性。最近 ELU 的变体和惩罚的 tanh 函数也表现出类似的性能改进。然而，ELU 与批量标准化（BN）方法之间的不相容性尚未得到很好的处理。

使用线性整流激活功能的神经元 ReLU 将输出：

$$f(\mathrm{x}) = \max(0, \mathrm{w}^{\mathrm{T}}\mathrm{x} + \mathrm{b}) \tag{2.2}$$

2.3.1.4 反向传导算法

对于单个样例(x, y)，那么其代价函数为：

$$\mathrm{J}(\mathrm{W}, \mathrm{b}; \mathrm{x}, \mathrm{y}) = \frac{1}{2}\left|\left|h_{W,b}(x) - y\right|\right|^2 \tag{2.3}$$

假设有一个固定样本集$\{x^{(1)}, y^{(1)}), \ldots, (x^{(m)}, y^{(m)})\}$，它包含 m 个样例，则整体代价函数为：

$$\begin{aligned}\mathrm{J}(\mathrm{W}, \mathrm{b}) &= \left[\frac{1}{m}\sum_{i=1}^{m} J(W, b); x^{(i)}, y^{(i)})\right] + \frac{\lambda}{2}\sum_{l=1}^{n_{l-1}}\sum_{i=1}^{s_l}\sum_{j=1}^{s_{l+1}}(W_{ji}^{(l)})^2 \\ &= \left[\frac{1}{m}\sum_{i=1}^{m}\left(\frac{1}{2}||h_{W,b}(x^{(i)}) - y^{(i)}\right)||^2\right] + \frac{\lambda}{2}\sum_{l-1}^{n_l-1}\sum_{i=1}^{s_l}\sum_{j=1}^{s_{l+1}}(W_{ji}^{(l)})^2\end{aligned} \tag{2.4}$$

目标是对于参数 w 和 b 来求函数$\mathrm{J}(\mathrm{W}, \mathrm{b})$的最小值，为了求解神经网络，将每一个参数$W_{ij}^{(l)}$和$b_i^{(l)}$初始化为一个接近 0 的、很小的随机值，之后对目标函数使用诸如批量梯度下降法的最优化算法。

在梯度下降中，每一次迭代都将更新参数 W 和 b，如下所示：

$$W_{ij}^{(l)} = W_{ij}^{(l)} - \alpha \frac{\partial}{\partial W_{ij}^{(l)}} J(W,b) \tag{2.5}$$

$$b_i^{(l)} = b_i^{(l)} - \alpha \frac{\partial}{\partial b_i^{(l)}} J(W,b) \tag{2.6}$$

α 是学习速率。

反向传播算法过程如下：

1. 前馈传导计算，利用前向传导公式，得到$L_2, L_3, \ldots$直到输出层的激活层。
2. 对于输出层，计算：

$$\delta^{(n_l)} = -\left(y - a^{(n_l)}\right) \cdot f'\left(z^{(n_l)}\right) \tag{2.7}$$

3. 对于$l = n_1 - 1, n_1 - 2, n_1 - 3, \ldots, 2$各层的，计算：

$$\delta^{(l)} = \left(\left(W^{(l)}\right)^T \delta^{(l+1)}\right) \cdot f'\left(z^{(l)}\right) \tag{2.8}$$

4. 计算最终需要的偏导数值：

$$\nabla_{W^{(l)}} J(W,b;x,y) = \delta^{(l+1)} (a^{(l)})^T \tag{2.9}$$

$$\nabla_{b^{(l)}} J(W,b;x,y) = \delta^{(l+1)} \tag{2.10}$$

2.3.2 朴素贝叶斯算法

朴素贝叶斯算法[10]根据事件的先验知识描述事件的概率，基于训练集 D 来估计类先验概率$P_{(c)}$，并为每个属性估计条件概率$P_{(x_i|c)}$。

令D_c表示训练集 D 中第 c 类样本的集合，如果存在足够的充足的独立同分布样本，则可容易地估计类的先验概率，

$$P_{(c)} = \frac{Dc}{D} \tag{2.11}$$

对离散属性而言，令$D_{(c,x_i)}$表示D_c在第 i 个属性上取值为x_i的样本组成的集合，则条件概率$P_{(x_i|c)}$可估计为，

$$P_{(x_i|c)} = \frac{|D_{c,x_i}|}{|D_c|} \tag{2.12}$$

对于输入向量$x = (x_1, \ldots, x_d) \in R_d$，机器学习分类旨在将 x 分类为离散可能类别集$\{c_1, \ldots, c_n\}$中的类。分类器由模型 W 和分类函数 CW 定义：

$R_d \to \{c_1, \ldots, c_n\}$。为了预测 x 所属的类，应该在 x 上计算函数 CW。输出$C_{i_0} \leftarrow CW_{(x)}$是分类结果。根据贝叶斯决策规则，朴素贝叶斯分类器的分类函数是选择具有最高后验概率的类。它应该计算$i_0 \leftarrow \arg\max_{i\in[m]} p(C=c_i \mid X=x) = \arg\max_{i\in[m]} p(C=c_i, X=x)$，其中因子 p（X＝x ）省略了恒定的 x。基于属性条件独立性假设，x 的每个属性（分量）是有条件独立的，每个类条件概率 p（C＝c_i，X＝x）等于分解 p（C＝c_i）$\Pi_{j=1}^{d}$（Xj＝xj|C＝c_i）。因此，对于朴素贝叶斯分类器，模型 W 由一组概率组成：先验概率{p（C＝c_1），...，p（C＝c_m）}，表示每个类

的发生概率c_i；类条件概率{{{p（X j=v|C=c_i）}v∈V_j} $_{j=1}^{d}$}$_{i=1}^{m}$表示 x 的第 j 个属性为 v 的概率 class c_i，其中 V j 是 X j 的域。

将D = $\{d_1, d_2, d_3, \ldots, d_p\}$视为一组文档，并且C = $\{c_1, c_2, c_3, \ldots, c_q\}$是类的集合。D 中的 p 个文档中的每一个被分类为来自集合 C 的 q 个数字类别之一。使用贝叶斯定理的文档 d 在类 c 中的概率由下式给出：

$$c_{map} = arg\max_{c\in C} P(c|d) = arg\max_{c\in C} \frac{P(c)P(d|c)}{P(d)} = P(c)P(d|c) \quad (2.13)$$

由于$P_{(d)}$与类无关，因此可以忽略它。

贝叶斯网络可以很好地表示复杂的概率分布，并且近年来已经得到了很多关注。朴素贝叶斯是贝叶斯网络的一个特例，贝叶斯分类代表了了监督学习技术和统计分类方法。在机器学习和统计中，分类是基于训练数据集识别新观察属于哪一类的问题。一个例子是将给定的电子邮件预测为垃圾邮件或非垃圾邮件，或者在给定的天气数据集中，能够预测是否会播放明天的天气情况。在机器学习方面，分类被计为监督学习的一个例子，即适当可预测观察的训练集的学习是可获得的。

2.3.2.1 朴素贝叶斯假设

假设属性彼此独立，在类 c 中文档 d 的贝叶斯概率计算如下：

$$P(d|c) = P(t_1|c)P(t_2|c)P(t_3|c)\cdots P(t_{n_d}|c) = \prod_{1\le k\le n_d} P(t_k|c) \quad (2.14)$$

替换c_{map}，得到下式：

$$\begin{aligned} c_{map} &= \arg\max_{c\in C} P(c\,|\,d) = \arg\max_{c\in C} P(c)P(d\,|\,c) \\ &= \arg\max P(c) \prod_{\substack{1\le k\le n_d \\ c\in C}} P(t_k\,|\,c) \end{aligned} \quad (2.15)$$

$P(c)$，c的先验概率计算如下：

$$P(c) = \frac{N_c}{N} \quad (2.16)$$

N_c是c类培训文件的数量，N 是培训文件的数量。$P(c\,|\,d)$被称为 c 的后验概率，因为它反映了在看到 d 之后 c 保持的置信度。

$$P(t\,|\,c) = \frac{T_{ct}}{\sum_{t'eV} T_{ct}} \quad (2.17)$$

其中T_{ct}是来自类 c 的 D 中出现的 t 的数量，并且是来自类 c 的 D 中的项的总数。

未出现在训练数据中的属性类组合将使整个结果为零。为了解决这个问题，使用 add-one 平滑或 Laplace 平滑。添加拉普拉斯校正后的等式变为：

$$P(t\,|\,c) = \frac{T_{ct}+1}{\sum_{t'eV}(T_{ct}+1)} = \frac{T_{ct}+1}{\sum_{t'eV}(T_{ct})+|V|} \quad (2.18)$$

根据定义将大量概率（介于 0 和 1 之间）相乘可能会导致浮点下溢。由于

$log_{(xy)} = log_{(x)} + log_{(y)}$，最好通过对概率的对数求和而不是乘以概率来执行所有计算。对具有最高最终非标准化对数概率分数的类而言，其概率是最高的。

已经提出了许多用于基本朴素贝叶斯算法的属性加权方法来减轻它们的属性独立性假设，适当的属性权重可以减少由于违反朴素贝叶斯中的属性独立性假设而导致的错误。显然，如果一组训练数据包括彼此相同的一组属性，则可以通过将总和为 1.0 的权重分配给集合中的属性集来消除由于违反属性独立性假设而导致的误差。例如，其中一个属性 Ai 的权重可以设置为 1.0，而与 Ai 相同的其余属性的权重设置为 0.0。这相当于从训练数据中删除剩余属性，即属性选择。属性权重为每个属性分配正的连续值权重，因此它比属性选择更灵活。属性选择可以被认为是属性加权的特定情况，其中权重值严格限制仅为 1.0 或 0.0。

2.3.2.2 特征加权朴素贝叶斯

特征加权方法根据预测特征的显着性对预测特征进行不同的加权，并且所得到的模型被称为特征加权朴素贝叶斯（FWNB）。 FWNB 将特征权重（代表特征的重要性）纳入公式中：

$$c(x) = \arg\max_{c \in C} P(c) \prod_{i=1}^{m} P(a_i \mid c)^{W_i} \tag{2.19}$$

其中 $W_i \in R^+$ 是第 i 个特征 Ai 的权重。

现在，唯一要回答的问题是如何定义每个预测特征的权重，这对于构建 FWNB 至关重要，并且越来越受到研究人员的关注。特征加权的研究是数据挖掘和机器学习中相对成熟的领域，并且大量现有的特征加权方法不利于详尽地呈现 FWNB。在这里，仅提供针对朴素贝叶斯专门设计的最先进的特征加权方法的研究调查。

最早为朴素贝叶斯设计的特征加权方法是 Ferreira 等人。然而，其实是为每个特征值而不是每个特征分配权重，因此不是严格的特征加权方法而是特征值加权方法。为了严格执行特征加权，Zhang 和 Sheng 首先提出了基于增益比的特征加权方法（GRFW），即具有更高增益比的特征值得更大的权重，因此将每个特征的权重设置为特征的增益比相对于所有特征的平均增益比。如下式所述：

$$W_i = \frac{GR(A_i)}{\frac{1}{m}\sum_{i=1}^{m} GR(A_i)} \tag{2.20}$$

其中 m 是特征的数量，$GR(A_i)$ 是使用特征 Ai 来划分给定训练实例的增益比，然后简单地将 Ai 的信息增益除以其分割信息。

Hall[76]提出了一种基于决策树的特征加权方法（DTFW）。在 DTFW 中，特征的权重与未压缩决策树中测试的最小深度成反比，然后通过对从训练数据中引

导 50%所产生的数据样本所获得的 10 个决策树求平均值稳定估计的权重。因此，DTFW 将权重分配给特征 Ai：

$$W_i = \frac{1}{T}\sum_{t=1}^{T}\frac{1}{\sqrt{d_{ti}}} \tag{2.21}$$

其中 d_{ti} 是在构建的未压缩决策树 t 中测试特征 Ai 的最小深度，T 是构建的决策树的总数。请注意，树的根节点具有深度 1，并且如果特征的权重未出现在构建的未修改的决策树中，则该特征的权重将设置为零。

Jiang[77] 等人提出了一种基于 Kullback-Leibler 测量的特征加权方法（KLMFW）。KLMFW 假设当观察到某个特征值 a_i 时，它向类变量 C 提供一定量的信息，然后使用 Kullback-Leibler 度量来计算特征值 a_i 的信息含量，如下：$\sum_c P(c\,|\,a_i)\log\frac{P(c\,|\,a_i)}{P(c)}$。然后可以将特征 Ai 的权重被定义为沿着 Ai 的所有特征值的 Kullback-Leibler 度量的加权平均值。为了保持其范围的真实性，学习的特征权重最终通过它们的均值来标准化。如下式所述：

$$\begin{aligned} W_i &= \frac{1}{Z}\sum_{a_i} P(a_i)KL(C|a_i) \\ &= \frac{1}{Z}\sum_{a_i} P(a_i)\sum_{c} P(c|a_i)\,log\frac{P(c|a_i)}{P(c)} \\ &= \frac{1}{Z}\sum_{a_i}\sum_{c} P(a_i)P(c|a_i)\,log\frac{P(c|a_i)}{P(c)} \\ &= \frac{1}{Z}\sum_{a_i}\sum_{c} P(a_i,c)\,log\frac{P(a_i,c)}{P(a_i)P(c)} \end{aligned} \tag{2.22}$$

其中 $Z = \frac{1}{m}\sum_{i=1}^{m} w_i$ 是归一化常数。尽管作者在论文中提出了两种实现方式，但是根据实验结果，上面没有拆分信息的特征加权呈现出最佳性能。

江[78]等人提出了一种深度特征加权方法（DFW）。DFW 使用基于相关的特征选择（CFS）滤波器从整个特征空间中选择最佳特征子集。然后为所选特征子集中的特征分配较大权重，为其他特征子集分配较小权重。为简单起见，将所选特征的权重设置为 2 和 1。如下式所述：

$$W_i = \begin{cases} 2, & \text{if } A_i \text{ is selected} \\ 1, & \text{otherwise.} \end{cases} \tag{2.23}$$

与所有其他现有的特征加权方法不同，DFW 是唯一将学习的特征权重不仅纳入朴素贝叶斯的分类公式而且还纳入其条件概率估计的。请注意，其将所提出的深度特征加权方法应用于一些最先进的朴素贝叶斯文本分类器，并且也取得了显着的改进。就目前所知，这种深度特征加权方法更适用于朴素贝叶斯文本分类器，如多项式朴素贝叶斯而不是标准的朴素贝叶斯。

可以看出，所有上述特征加权方法在运行特征加权朴素贝叶斯之前根据基于通用数据特征的启发直接计算特征权重，因此是特征加权滤波器。除了它们之外，还存在另一类特征加权方法，其使用来自特征加权朴素贝叶斯的性能反馈来整体地优化特征权重，其称它们为加权包装器。

2.4 网络流量分类统计特征

网络流量分类统计特征包括主机层特征和网络流层特征。主机层统计特征是指以主机为统计单位计算得到的特征，如单位时间内的连接数统计量、连接到达时间统计量等。网络流层特征是指以网络流为统计单位计算得到的特征，如网络流持续时间、网络流字节统计量等。

上述统计量又可以进一步细分为分位数、百分位数、最大值、最小值、均值、方差、标准差以及中位数等。分位数、百分位数、中位数都是针对有序序列而言，需要先对一组统计值进行排序。分位数如分别取第一个 4 分位值（即有序序列 1/4 位置对应的值）和第三个 4 分位值（即有序序列 3/4 位置对应的值），25%百分位数和 75%百分位数分别对应第一个 4 分位值和第三个 4 分位值，中位数为有序序列中间位置对应的值。

Moore[79]等人提出了 249 个可用于网络流分类的统计特征，包括了当前流量分类研究中使用的绝大多数特征。这些特征大致可分为双向特征和单向特征，设通信双方分别用客户端 C 和服务器 S 表示，C 和 S 之间的一个流由 C 发往 S 的全部数据包，记为$C \rightarrow S$，和 S 发往 C 的全部数据包，记为$S \rightarrow C$组成。双向特征是基于$C \rightarrow S$ 和 $S \rightarrow C$两个方向汇合统计，单向特征基于$C \rightarrow S$或$S \rightarrow C$中某一个方向分别统计。

双向特征包括服务器和客户端所用端口号、数据包到达间隔时间统计量、以太帧字节长度统计信息、IP 包字节长度统计量和数据包中控制字节的统计量。

单向特征包括传输的字节总数、发送的各类数据包计数（包括 ACK 包、纯 ACK 包、SACK 包、D-SACK 包、PUSH 包、SYN 包、FIN 包、URG 包等）、发送的字节数、TCP 数据载荷包计数、TCP 数据载荷字节计数、重传包计数、重传字节计数、窗口探测包计数、窗口探测字节计数、窗口参数相关统计、失序包计数、分片包长统计量、数据传输时间、空闲时间、吞吐量、RTT 相关统计量、分片应答计数、包重传时间统计量、以太帧字节长度统计量、IP 包字节长度统计量、数据包中控制字节的统计量、包到达间隔时间统计量、事务模式和块传输模式下的事务数、块传输模式时间量及占比、连续持续时间、空闲时间量及占比、基于熵的有效带宽、包到达间隔时间统计量的快速傅立叶变换值等。

第 3 章 加密网络流量数据集

本章介绍了网络流量加密协议 SSL/TLS 的工作流程。加密网络流量的收集方法，加密网络流量存储的文件格式 P 和用于加密网络流量数据集的预处理流程。

3.1 SSL/TLS 协议简介

SSL[80]加密协议(Secure Socket Layer，安全套接层)和它的继任者 TLS[81]加密协议（Transport Layer Security，传输应用层），是为计算机网络间的交流提供安全保障。这些协议广泛应用在网页浏览，邮件，即时通信，音频电话等应用。SSL/TLS 协议位于 TCP/IP 协议和各种应用层协议之间，加密和传输数据，并且为数据通讯提供安全支持。SSL/TLS 包含握手协议和记录协议，在握手协议中，使用数字证书保证了公钥的安全性，如图 3-1 完整的 TLS 握手协议报文流所示。另一个使用握手协议的参数来转换加密数据的记录协议。

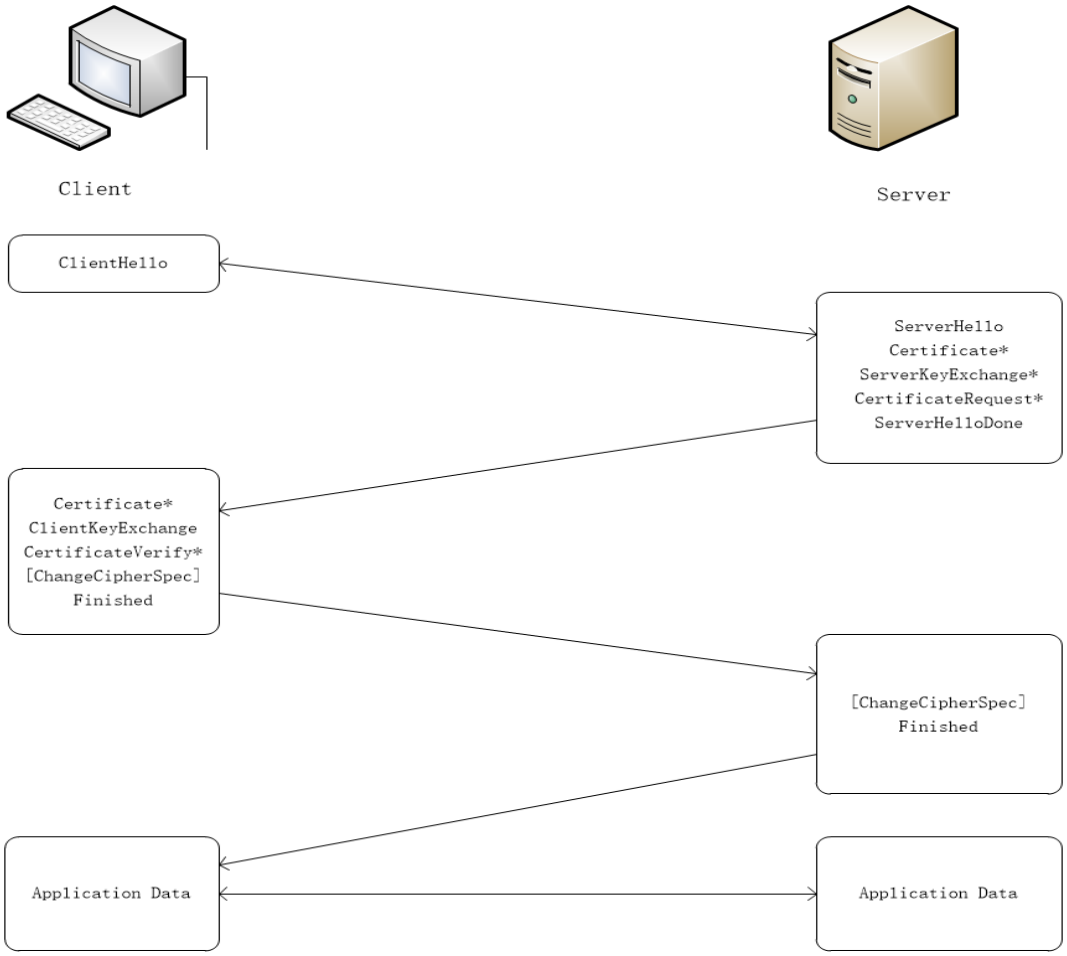

图 3-1 完整的 TLS 握手协议报文流

3.2 加密流量采集

实验数据集采用 ISCX VPN-nonVPN[82]数据集，包含经由 OpenVPN 通信产生的各类流量。UNB ISCX 网络流量（VPN-nonVPN）数据集由标记的网络流量组成，包括 pcap 格式的完整数据包和 csv（ISCXFlowMeter 生成的流量）也可供研究人员公开使用。该数据集使用 Wireshark 和 Tcpdump 捕获流量，生成了总共 28GB 的数据。为了生成 SFTP 和 FTPS 流量，还使用外部服务提供商和 Filezilla 作为客户端。

Tcpdump 是一种通用的网络数据包捕获和分析工具，可以在大多数类 Unix 平台上运行。在 Linux、Salaris、BSD、HP-UX 和 AIX 等系统上，Tcpdump 基于 Libpcap 库实现捕包功能。Tcpdump 是一个命令行工具，可以实现网络捕包、规则过滤、文件输出等输出。

Wireshark 是运行于 Windows 及类 Unix（包括 Linux）平台上的一个具有图形化用户接口的包捕获和分析工具，其前身为著名的 Ethereal 软件。Windows 版的 Wireshark 基于 Winpcap 开发。Wireshark 的另一个特点是采用了 PCAP 文件格式的拓展版本 pcap-ng。pcap-ng 文件格式扩展名为.pcapng，支持存储更多信息，例如更细的时间戳精度、捕包接口信息、捕包统计信息等，可以用 tcpdump 命令将 pcap-ng 格式的文件转换成 PCAP 格式的文件。图 3-2 是 Wireshark 工具展示的用户界面，在 Wireshark 中可以直观地显示帧级源 MAC 地址和目的 MAC 地址、IP 头部、TCP 头部的有关信息。

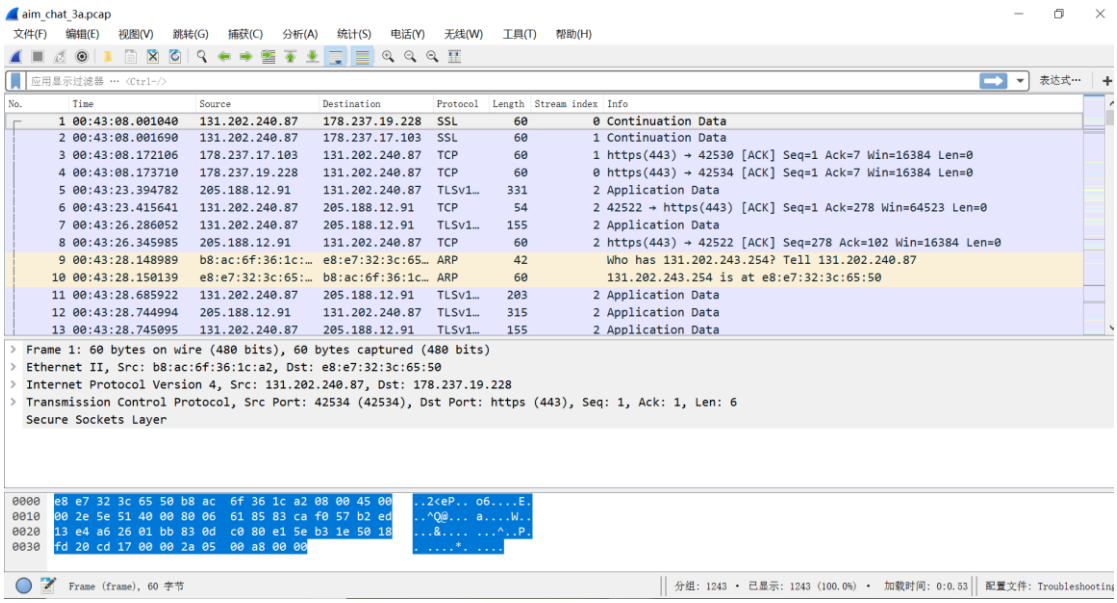

图 3-2　Wireshark 用户界面

3.2.1 网络数据集 PCAP 文件格式

网络数据包捕获工具 Wireshark 采用 PCAP 文件格式来存储数据。PCAP 文件是二进制文件，包含文件头和数据记录两部分，文件头记录了该 PCAP 文件的

一些属性信息，数据部分是按包捕捉时间逐个记录的数据帧记录。PCAP 文件的详细格式如图 3-3 所示.

图 3-3　PCAP 文件格式

PCAP 文件头部分如图 3-4 所示，最前 32 位是文件类型标记，主、副版本号各 16 位，区域时间和精确时间戳各 32 位，大多数情况下不用，均置为 0。捕包长度用来设置数据包的捕获长度（包含链路层帧头和 IP 头）。链路类型字段给出了一些典型链路的预设值，其中最常用的值为以太网 0x0001。

图 3-4　PCAP 文件头

在 PCAP 文件头之后，是按数据包捕获时间记录的一系列数据包。每个数据包包含两个部分，PCAP 包头和 IP 数据帧。PCAP 包头共 16 字节，如图 3-5 所示，由三部分组成。首先是时间戳 8 字节，structtimeval 结构，该结构由两个 4 字节的成员组件组成，高 4 字节存储秒计时，记录的是自格林威治时间 1970 年 1 月 1 日 0 时 0 分 0 秒到该包捕捉时间（系统时间）之间以秒计算的数值。低 4 字节是捕获该数据包时的微秒值，即 PCAP 捕包理论上可以精确到微秒。在时间戳后，用 4 字节表示捕获该 IP 数据帧时的长度，而后再用 4 字节表示该 IP 数据帧的实际长度。

图 3-5　PCAP 包头

3.2.2 数据集流量类别

为了在 ISCX 中生成实际流量的代表性数据集，定义了一组任务，同时为了确保采集的数据集在多样性和数量上足够丰富，创建了用户 Alice 和 Bob 帐户，以便使用 Skype，Facebook 等服务。

通过 VPN 捕获了常规会话和会话，共有 14 个流量类别：VOIP，VPN-VOIP，P2P，VPN-P2P 等。以下为所生成的不同类型的流量：

Browsing：在此类别下，在浏览或执行包括使用浏览器的任务时，会生成用户生成的 HTTPS 流量。例如，当使用 Hangouts 捕获语音呼叫时，即使浏览不是主要活动，也会捕获多个浏览流程。

Email：使用 Thunderbird 客户端以及 Alice 和 Bob Gmail 帐户生成的流量示例。客户端配置为通过 SMTP/S 传递邮件，并在一个客户端中使用 POP3/SSL 接收邮件，在另一个客户端中使用 IMAP/SSL 接收邮件。

Chat：聊天类标识即时消息应用程序。使用名为 pidgin 的应用程序通过网络浏览器，Skype，IAM 和 ICQ 进行 Facebook 和 Hangouts。

Streaming：网络流类标识需要连续和稳定数据流的多媒体应用程序。使用 Chrome 和 Firefox 从 Youtube（HTML5 和 Flash 版本）和 Vimeo 服务中捕获了流量。

File Transfer：此类别标识主要用于发送或接收文件和文档的流量应用程序。在数据集种，捕获了 Skype 文件传输，FTP over SSH（SFTP）和 FTP over SSL（FTPS）流量会话。

VoIP：IP 语音标签将语音应用程序生成的所有流量分组。在此标签中，使用了 Facebook，Hangouts 和 Skype 等应用程序用来捕获语音通话。

P2P：此类别用于标识文件共享协议，如 Bittorrent。为了生成此流量，从公共存储库下载了不同的.torrent 文件，并使用 uTorrent 和 Transmission 应用程序捕获了流量会话。

UNB ISCX 网络流量数据集内容如表 3-1 所示：

表 3-1 UNB ISCX-网络流量数据集内容

Traffic	Content
Browsing	Firefox and Chrome
Email	SMPTS，POP3S and IMAPS
Chat	ICQAIM，Skype，Facebook and Hangouts
Streaming	Vimeo and Youtube
File Transfer	Skype，FTPS and SFTP using Filezilla and an external service
VoIP	Facebook，Skype and Hangouts voice calls (1h duration)
P2P	uTorrent and Transmission (Bittorrent)

3.3 加密流量数据集预处理

本文使用经由 OpenVPN 通信产生的 7 类流量，数据集目录及数目如表 3-2 数据集目录及数目统计所示。包含以下 7 种类型的流量。

Browsing：用户在浏览器使用 https 协议浏览网页时生成的网络流量。

Email：用户收发邮件时生成的网络流量。

Chat：用户使用即时通讯软件时生成的网络流量。

Streaming：用户使用多媒体应用时生成的网络流量。

File Transfer（简称为 FT）：用户发送和接受文件生成的网络流量。

VoIP: 语音应用生成的网络流量。

P2P：识别类 Bittorrent 的文件分享协议。

表 3-1 数据集目录及数目统计

流量类别	数目
Browsing	1621
Email	569
Chat	4546
Streaming	1144
FT	1794
VoIP	11008
P2P	709

3.3.1 流量预处理流程

在获取 Wireshark 捕获的数据包后，运用数据预处理工具 ISCXFlowMeter 工具生成网络流并提取网络流基于时序的特征，而网络流由一系列序列{源 IP，目的 IP，源端口，目的端口，网络协议}相同的数据包组成。

在以往的许多研究中都使用了一种 NetMate 的网络流预处理工具，网络流由一系列序列{源 IP，目的 IP，源端口，目的端口，网络协议}相同的数据包组成。本文将采用 ISCXFlowMeter 生成网络流并提取流的基于时序的特征，ISCXFlowMeter 运用 Java 编写，并能更方便、灵活的统计网络流的各种特性，还能很好的控制流量超时的持续时间。ISCXFlowMeter 能生成双向流，第一个数据包决定了流的方向是向前（源到目的地）还是向后（目的地到源），分别统计流在向前、向后的时间相关的特征。TCP 流通常通过 FIN 包判断断开连接，而 UDP 流由超时终止判断断开连接。

3.3.2 网络流特征提取

在选择与时序相关的特征时，考虑两种不同的方法。第一种方法通过检测时间，例如流量包的到达时间间隔或网络流的存活时间。第二种方法，固定时间不变，同时检测其他变量，例如每秒字节数或者每秒数据包数。

表 3-3 提供了完整的时序相关的特征提取的列表。从表 3-3 可以看出，除了 Duration 显示网络流的持续时间，有六组特征。前三组是：-fiat、-biat 和-flowiat，分别集中在前向，向后和双向流动。第四组和第五组特征，是关于计算空闲到活动或活动到空闲状态并命名为-idlea 和-active。

表 3-3 时序相关的特征列表

特征	描述
duration	流的持续时间
fiat	向前数据包到达间隔时间
biat	向后数据包到达间隔时间
flowiat	任意方向的数据包到达间隔时间
active	流的活跃时间
idle	流的空闲时间
fb_psec	每秒流的比特数
Fp_psec	每秒流的数据包数目

3.3.3 数据规范化

数据归一化是指将特征值按比例缩放到特定的区间内以便于数据处理和加快程序收敛，本文将特征值缩放到[0.0，1.0]，采用最小-最大规范化方法。

最小-最大规范化方法通过线性变换将原始数据映射到[0.0，1.0]这个区间中，若设特征的最大值和最小值分别为x_{max}和x_{min}，则映射函数为

$$x' = \frac{x - x_{min}}{x_{max} - x_{min}} \tag{3.1}$$

最小-最大规范化方法的优点是保留了原始数据之间的顺序关系，但是需要事先知道该特征的最大值和最小值，并且一旦新数据未落在$[x_{min}, x_{max}]$区间内，则会产生越界错误。当特征的数据值均大于等于 1 时，可以利用映射空间

$$x' = \frac{\log_{10} x}{\log_{10} x_{max}} \tag{3.2}$$

将原始数据映射到[0.0，1.0]空间上。

第 4 章 基于深度神经网络的加密流量分类方法

本章主要研究对经由 VPN 传输的多样化的网络流量分类，将运用深度学习中的深度神经网络构建加密流量分类模型。本章定义了深度神经网络分析模型的结构，提出了分类算法，分析了算法的时间复杂度，叙述了分类模型的训练过程，并提出了可能影响深度神经网络分类模型分类能力的因素 mini-batch size。与传统的流量分类方式相比，基于深度神经网络的加密流量分类可以更好的对多种网络流量进行分类，同时无需人工挑选特征，简化了分类流程，能更好的适应当前网络流量来源广泛、种类多样的现状。

4.1 基于深度神经网络的加密流量分类模型设计

基于深度神经网络的加密流量分类模型主要是通过训练深度神经网络，同时根据深度神经网络的分类结果与应得到的结果的之间的误差更新各个神经元的参数，从而使深度神经网络具有对多种网络流量的分类能力。

4.1.1 基于深度神经网络的加密流量分类模型整体架构

基于深度神经网络的加密流量分类主要流程为，通过 Wireshark 采集经由 OpenVPN 传输的网络数据包，网络数据包经过 ISCXFlowMeter 生成网络流，提取网络流中基于时序的特征生成向量。然后将用户向量划分为训练集和测试集，接着运用训练集的数据训练并生成基于深度神经网络模型，最后运用生成的分类模型预测测试集中加密流量的类别。图 4-1 显示了基于深度神经网络的加密流量分类模型的整体架构。

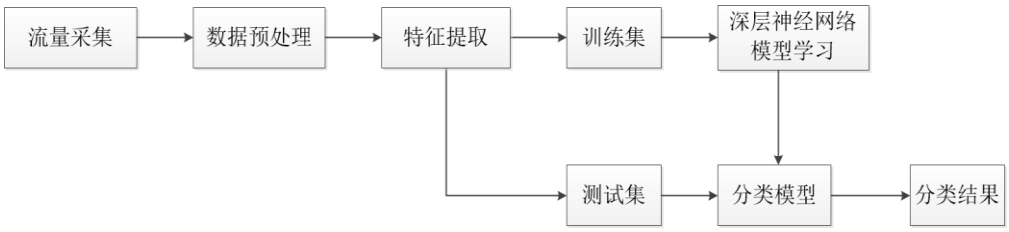

图 4-1 基于深度神经网络的加密流量分类模型的整体架构

网络流量采集是指使用 Wireshark 捕获经由 OpenVPN 传输的网络流量数据包，网络流量数据包信息主要包括源 IP 地址、目的 IP 地址、源端口、目的端口、网络协议、用户数据等信息。数据预处理是提取网络数据流，将具有相同{源 IP 地址、目的 IP 地址、源端口、目的端口、网络协议}的数据包提取为网络数据流。特征提取是指提取网络数据流中基于时序的网络流量特征，例如数据包到达的间隔时间等。特征预处理是指对网络流量特征数据进行规范化处理，并将特征值按比例缩放到特定的区间内[0.0，1.0]，目的是方便数据处理及加快程序收敛。

4.1.2 接口设计

基于深度神经网络的加密流量分类模型主要接口包括网络流数据读取、数据集划分、深度神经网络、深度神经网络参数接口、数据分析。

网络流数据读取接口主要从 csv 文件读取经过预处理后的网络流数据。

数据集划分接口是指将网络流量数据集按照 7：3 的比例随机划分成训练集和测试集。

深度神经网络参数接口，深度神经网络在训练过程中的参数学习是基于梯度下降法进行优化的，而基于梯度下降的优化算法需要给每个参数一个初始值。

深度神经网络接口主要规定了深度神经网络的主体结构，包括各层的神经节点数，隐层的数目等，输入和输出的规范，激活函数的选取、优化算法的选取，mini-batch size 的设置等。

数据分析接口主要是将深度神经网络分类模型的预测结果与应有结果进行对比，并计算深度神经网络分类模型的分类能力。

4.1.3 实验框架

本文采用的实验框架主要有 Tensorflow[83]、Numpy 和 Scikit-learn。

深度学习框架 Tensorflow 是由 Google Brain 团队开发的用于数值计算的开源软件库。当前，Tensorflow 广泛用于机器学习和神经网络方面的研究，Tensorflow 架构灵活，可以在多种平台展开计算，能自动求微分，简化计算，支持多语言，能使计算机性能最优化。本文主要使用的 Tensorflow 模块如表 4-1 所示。

表 4-1 Tensorflow 模块

名称	功能
tf.Variable	参数初始化
tf.random_normal	从服从指定正态分布的数值中取出指定个数的值
tf.add	加法计算
tf.nn.relu	计算激活函数 relu，即 max(x，0)
tf.matmul	矩阵乘法
tf.argmax	对象在某一维上的其数据最大值所在的索引值
tf.equal	判断两个元素是否相等
tf.reduce_mean	用于计算张量的平均值
tf.cast	转换数据格式
tf.placeholder	占位符
tf.nn.softmax_cross_entropy_with_logits	计算交叉熵

Numpy 是 python 语言的一个拓展程序库，支持大量的维度数组与矩阵运算，同时为数组运算提供了大量的数学函数库，表 4-2 为本文主要用到的 Numpy 模块。

表 4-2　Numpy 模块

名称	功能
numpy.array	数组运算
numpy.nan_to_num	使用 0 替代数组中的 NaN，使用有限数字代替 infinity

Scikit-Learn 是一个用于 Python 编程语言的免费机器学习软件库，具有各种分类、回归和聚集算法，包括支持向量机、随机森林、梯度增强、K-means 和 DBSCAN，旨在与 Python 数值计算库 Numpy 和 Scipy 相互协作。表 4-3 为本文主要用到的 Scikit-learn 模块。

表 4-3 Scikit-Learn 模块

名称	功能
train_test_split	按照一定的比例，将数据集随机划分为训练集和测试集

4.2 深度神经网络结构

4.2.1 深度神经网络结构

在基于深度神经网络的加密流量分类模型中，第一层为输入层，输入数据为预处理后的网络流基于时序的特征。第 2 层到第 7 层为 6 个隐层，每一个隐层都由 256 个神经网络节点组成。

设有m个隐层，L_1，L_2，…，L_m，每个隐层有 n 个神经元，L_l 层中的神经元为 $S_1^{(1)}$，$S_1^{(2)}$，…，$S_1^{(n)}$，

则第一个隐层的第i个神经元的输出为，

$$S_1^i = V\left(\Theta_1^{(i,0)}x_0 + \Theta_1^{(i,1)}x_1 + \cdots + \Theta_1^{(i,n)}x_n\right), \quad \mathrm{i} = 1,2,\ldots,\mathrm{m} \tag{4.1}$$

其中，Θ_1是一个参数矩阵，x是输入的特征向量，V为ReLU激活函数，ReLU是一种常用的激活函数，运用仿生物学原理，计算方便，同时它不会像 sigmoid 激活函数一样在浅梯度上饱和[84]。

$$\mathrm{V(t)} = \max(0,\mathrm{t}) \tag{4.2}$$

其中，t 为激活函数的输入。

在获得了第一个隐层的输出后，可以向前传播到下一个隐层，直到最后一层。第$\mathrm{j}+1$层隐层的第i个神经元的输出为：

$$S_{j+1}^i = V\left(\Theta_{j+1}^{(i,0)}S_j^0 + \Theta_{j+1}^{(i,1)}S_j^1 + \cdots + \Theta_{j+1}^{(i,n)}S_j^n\right), \quad \mathrm{i} = 1,2,\ldots,m \tag{4.3}$$

最后一层为输出层，输出层预测网络流的类别，根据输出向量中最大值的索引号，确定输出的类别。

输出层输出结果可由最后一个隐层计算得到，

$$S_{out}^{i} = V\left(\Theta_{out}^{(i,0)} S_m^0 + \Theta_{out}^{(i,1)} S_m^1 + \cdots + \Theta_{out}^{(i,n)} S_m^n\right), \quad \mathrm{i} = 1,2,\ldots,\mathrm{m} \quad (4.4)$$

获得了输出层的值后，可以计算损失函数，损失函数是用来估算模型的预测值 S 与真实值 y 的不一致程度，是一个非负实值函数，损失函数越小，模型的鲁棒性就越好，本文采用交叉熵损失函数。预测值与实际值的差值J：

$$\mathrm{J} = -\frac{1}{m}\left[\sum_{i=1}^{m}\sum_{k=1}^{7} y_k^{(i)} \log\left(S(x^i)\right)_k + \left(1 - y_k^{(i)}\right) log\left(1 - \left(S(x^{(i)})\right)_k\right] \quad (4.5)$$

其中，m为训练样本的数目，$y_k^{(i)}$表示第i个样本的类别，如果第i个样本属于第k种类别，$y_k^{(i)} = 1$，否则，$y_k^{(i)} = 0$，x^{i}表示训练集的第i个样本，$(S(x^i))_k$是对于输入$x^{(i)}$的第k个输出单元的值。Θ为$(S(x^i))_k$函数的主要参数，在深度神经网络算法中，可以通过优化Θ使得损失函数 J 最小化。

前向传播算法

Input：网络深度 l，$W^{(i)}, i \in \{1, \ldots., l\}$，模型的权重矩阵，$b^{(i)}, i \in \{1, \ldots., l\}$，模型的偏置参数，x，程序的输入

Output：y，程序的输出

$h^{(0)} = x$

for $k = 1, \ldots, l$ do

$a^{(k)} = b^{(k)} + W^{(k)} h^{(k-1)}$

$h^{(k)} = f(a^{(k)})$

end for

$\mathcal{y} = h^{(l)}$

$J = L(\mathcal{y}, y) + \lambda\Omega(\theta)$

4.2.2 深度神经网络算法训练

在完成基于深度神经网络的加密网络流量分类模型设计后，需要训练深度神经网络模型。

深度神经网络的具体训练过程如下：

1．定义神经网络，主要包括设置输入数据和输出数据的格式，设置隐藏层的层数和每层神经元的数目，设置神经元的函数、激活函数，设置学习率，设置batch_size，设置最大训练次数和期望误差。

2．定义误差计算函数，本文为交叉熵损失计算函数，设置优化算法为 Adam

3．输入训练样本，并随机打乱。

4．计算样本的期望输出值与实际输出值之间的误差。

5．根据误差运用反向传播算法计算并更新深度神经网络分类模型中的各个神经元参数。

6．重复第 3 步到第 5 步，直到误差小于或等于期望误差，或者达到最大训练次数。

如上述步骤生成并训练深度神经网络分类模型，更新深度神经网络神经元参数，得到基于深度神经网络的加密网络流量分类模型。此后可以用此模型对预测集的样本预测并分类。

4.2.3 Adam 优化算法

深度神经网络在训练过程中会逐渐更新很多神经元的参数，直到能正确把输入信号映射到分类结果。误差与权重之间的关系为$\frac{dE}{dw}$，衡量的是权重调整后误差的变化程度。

微积分的链式法则公式为：

$$\frac{\mathrm{dz}}{\mathrm{dx}}=\frac{dz}{dy}\cdot\frac{dy}{dx} \tag{4.6}$$

在深度神经网络中，误差、权重和激活函数之间的关系如下所示：

$$\frac{dError}{dweight}=\frac{dErroe}{dactivation}*\frac{dactivation}{dweight} \tag{4.7}$$

Adam 优化算法

Input: α：步长， $\beta_1,\beta_2\in[0,1)$: 目前估计的指数衰败率， f(θ): 带有参数 θ 的随机目标函数， θ_0初始参数向量

Output: θ_t

$m_0\longleftarrow 0$ (初始化第一时刻向量)

$v_0\leftarrow 0$（初始化第二时刻向量）

$\mathrm{t}\leftarrow 0$（初始化时长）

while θ_t 没有聚合 do

$\mathrm{t}\leftarrow \mathrm{t}+1$

$g_t\leftarrow\nabla_\theta f_t(\theta_{t-1})$ (在时长 t 获得 w.r.t 随机目标梯度)

$m_t\leftarrow\beta_1\cdot m_{t-1}+(1-\beta_1)\cdot g_t$ (有偏一阶矩估计更新)

$v_t\leftarrow\beta_2\cdot v_{t-1}+(1-\beta_2)\cdot g_t^2$ (有偏一阶矩估计更新)

$\widehat{m}_t\leftarrow m_t/(1-\beta_1^t)$ (一阶矩的偏差修正)

$\hat{v}_t\leftarrow v_t/(1-\beta_2^t)$ (二阶矩的偏差修正)

$\theta_t\leftarrow\theta_{t-1}-\alpha\cdot\widehat{m}_t/\left(\sqrt{\hat{v}_t}+\epsilon\right)$ (更新参数)

end while

本文采用 Adam[85]优化算法来优化参数Θ，Adam 是一种可以替代传统随机梯度下降过程的一阶优化算法，能基于训练数据迭代的更新神经网络权重。Adam 的计算效率高，所需的内存少，适合解决含有大量数据和参数的优化问题。

4.2.4 时间复杂度分析

对于深度神经网络分类算法，假设 N 是训练样本的数目，d 是每个样本的维数，H 是隐层的数目，i是隐层神经元的数量，则深度神经网络的时间复杂度是

$$O(Ndi + N(H-1)i^2 + Ni) \tag{4.8}$$

4.3 mini-batch 下降法

当每次对整个训练集的数据进行梯度下降时，为 batch 梯度下降。每次参数更新时，深度神经网络遍历整个训练集计算损失函数，再计算函数对每个参数的梯度，然后更新各个参数。这种方法每更新一次参数都要把所有样本计算一次，数据计算的代价大，并且计算速度慢，也不支持在线学习，这被称批梯度下降（batch gradient descent）。

当每次是计算一个样本就进行一次梯度下降计算的时候，就是 stochastic 梯度下降。每计算一个样本就计算一次损失函数，然后根据误差计算梯度同时更新参数，这被称为随机梯度下降（stochastic gradient descent）。优点是参数更新速度快，缺点是收敛性能不太好，极有可能在最优点附近徘徊，而最终无法到达最优点。更为严重的是，两次参数的更新很有可能会彼此抵消掉，造成目标函数震荡剧烈，从而无法达到优化深度神经网络的目的。

当每次处理的样本数量在一个样本和所有样本之间时，被称为小批量梯度下降（mini batch 梯度下降，）。Mini batch 下降法克服了批梯度下降和随机梯度下降法的缺点。Mini batch 下降法把数据分成若干批，每次计算一批数据，按照一批的误差来更新深度神经网络的神经元参数。采用这种计算方式，同一个批中的所有数据共同决定了本次梯度下降的方向，下降时不容易跑偏，极大地减少了随机性。与此同时，因为每次需要计算误差地样本数与整个数据集相比小了很多，计算量不会太过庞大。与批梯度下降相比，使用小批量梯度下降地方法更新深度神经网络地参数更快，而且有利于更鲁棒地收敛，并且可以避免局部最优。而与随机梯度下降法相比，使用小批量梯度下降法的计算效率更高，训练模型的速度可以加快。

Goyal[86]发现 mini-batch SGD 能有效提高模型学习的效率。Mini-batch 是指从训练集随机挑选小的子集来训练，这些子集被称为 mini-batch，每个子集内样本的数目被称为 mini-batch size，假设每个 mini-batch size 为 m，样本数为 x，则 1<m<x，则优化后的参数Θ'为：

$$\Theta_{\mathrm{i}}' = \Theta_i - \alpha \sum_{j=t}^{t+m-1} (S_\Theta(x_0^j, x_1^j, \dots x_n^j) - y_j)x_i^j \tag{4.9}$$

小批量计算时随机抽取也非常重要。从一组样本中计算出梯度期望的无偏估计要求这些样本是独立的。

本文将探讨当 mini-batch size 取不同的值时，对模型分类能力的影响。

深度神经网络分类模型算法

Input：训练集$D = \{(x_i, y_i)\}_{i=1}^{n}\}$，learning rate η，epochs ，mini-batch size s, 初始化网络权值 Θ

Output：网络权值 Θ

```
for j in range(epochs):
    random.shuffle(D)
    mini-batches = [D[k:k+s]]
    for k in range(0,n,s):
        for mini-batch in mini-batches:
            loss = J(mini_batch_y, Θ, mini − batch_x)
            Adam(η, loss)
            Update(Θ)
        end for
end for
end for
```

第 5 章 基于深度神经网络的加密流量分类实现

本章介绍了实验环境，训练并检验了基于深度神经网络的加密网络流量分类模型的分类能力。探讨了 mini-batch size 的变化对基于深度神经网络模型的网络流量分类模型的分类能力的影响。同时，将本文提出的基于深度网络的网络流量分类模型与朴素贝叶斯算法进行了对比，研究了两者的加密网络流量分类能力。

5.1 实验环境

深度神经网络算法采用 Python 语言编写，第三方软件及 API 包括：Pycharm，Tensorflow，numpy，sklearn。朴素贝叶斯算法运用 Weka 实现。主机的配置为：Dell XPS13，CPU 为 Intel Core i5-8250U 1.6GHZ，内存 8G，硬盘 225GB；操作系统为 win10 家庭中文版。

5.2 实验结果与分析

5.2.1 实验评价指标

本文使用两个评价指标：分类准确率(Precision)和分类召回率（Recall）来评判分类算法的性能。

$$\text{Precision} = \frac{\text{TP}}{\text{TP} + \text{FP}} \tag{5.1}$$

$$\text{Recall} = \frac{\text{TP}}{\text{TP} + \text{FN}} \tag{5.2}$$

TP 是被正确分类为该类的数量，FP 是被错误划分为该类的数量，FN 是被错误划分为非该类的数量。

5.2.2 实验结果分析

图 5-1~5-7 为基于深度神经网络加密流量分类模型对各个类别随着 mini-batch size 变化时的分类准确率与分类召回率。

图 5-1 为基于深度神经网络加密流量分类模型对 VoIP 类的分类准确率与分类召回率随着 mini-batch size 变化的统计图。在图中，随着 mini-batch size 的变化，VoIP 类的分类准确率在 0.69~0.86 之间变化，分类召回率在 0.88~0.97 之间变化，但分类召回率始终高于分类准确率。当 mini-batch size 为 60 时，分类准确率最低为 0.69，但分类召回率却高达 0.97，可以看出 mini-batch 的取值变化对 VoIP 的分类准确率和分类召回率产生了不同的影响。

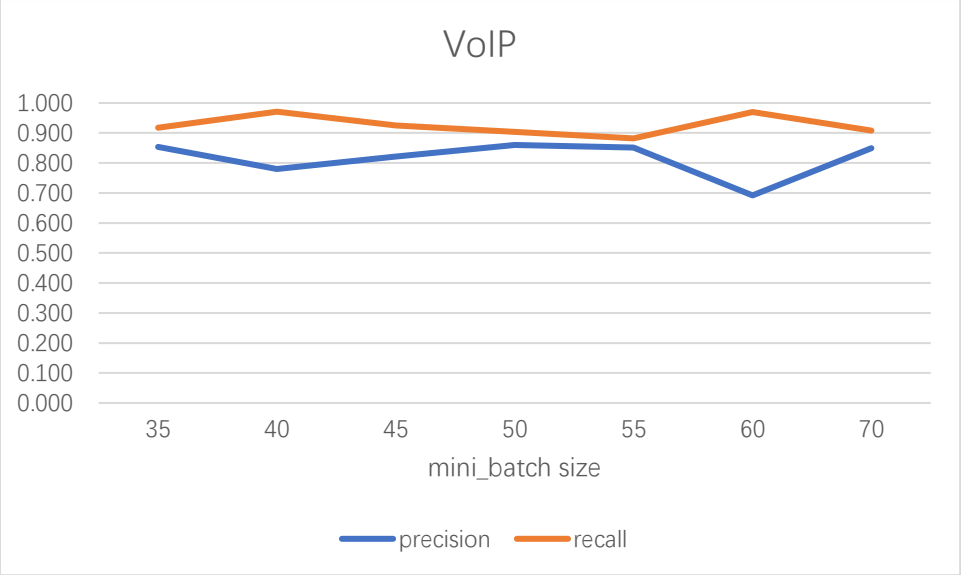

图 5-1　VOIP 类分类准确率与分类召回率

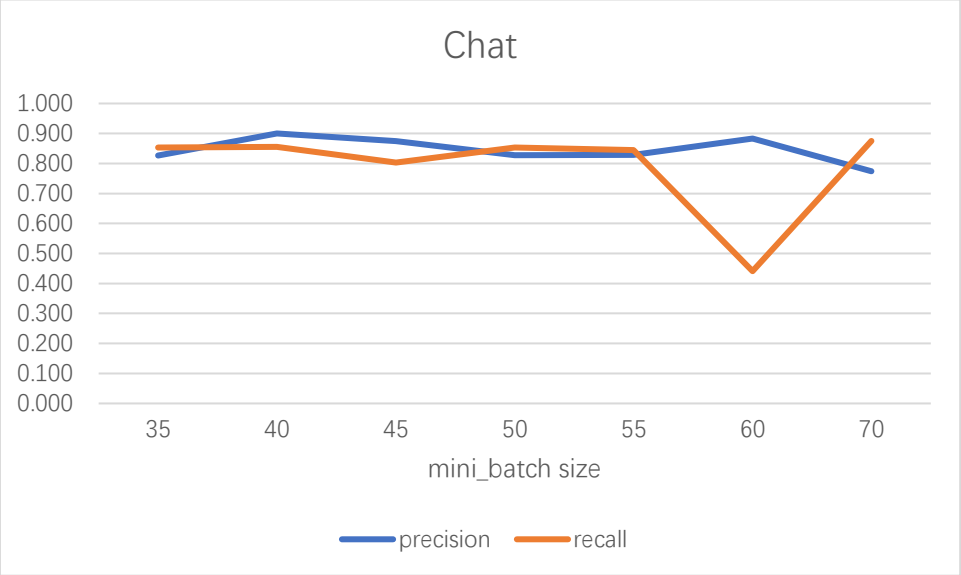

图 5-2　Chat 类分类准确率与分类召回率

图 5-2 为基于深度神经网络加密流量分类模型对 Chat 类的分类准确率与分类召回率随着 mini-batch size 变化的统计图。在图中，基于深度神经网络加密流量分类模型对 Chat 类的分类准确率在 0.77~0.90 之间变化，分类召回率在 0.44~0.88 之间变化，分类召回率受 mini-batch size 变化明显比分类准确率大。当 mini-batch size 为 40 时，分类准确率取得最大值 0.9。mini-batch size 为 60 时，分类召回率取得最低值为 0.44.

图 5-3 为基于深度神经网络加密流量分类模型对 FT 类的分类准确率与分类召回率随着 mini-batch size 变化的统计图。在图中，FT 类的分类准确率与分类召回率相比，受到 mini-batch size 变化的影响更大。在图中，FT 类分类准确率在 0.48~0.79 之间变化，而分类召回率在 0.45~0.61 之间变化。当 mini-batch size 为 40 时，分类准确率取得最大值 0.79，分类召回率却取得最低值 0.45。当 mini-batch

size 为 45 时，分类准确率却取得最低值 0.48，分类召回率取得最高值 0.61。mini-batch size 的变化对分类准确率和分类召回率的影响正好相反。

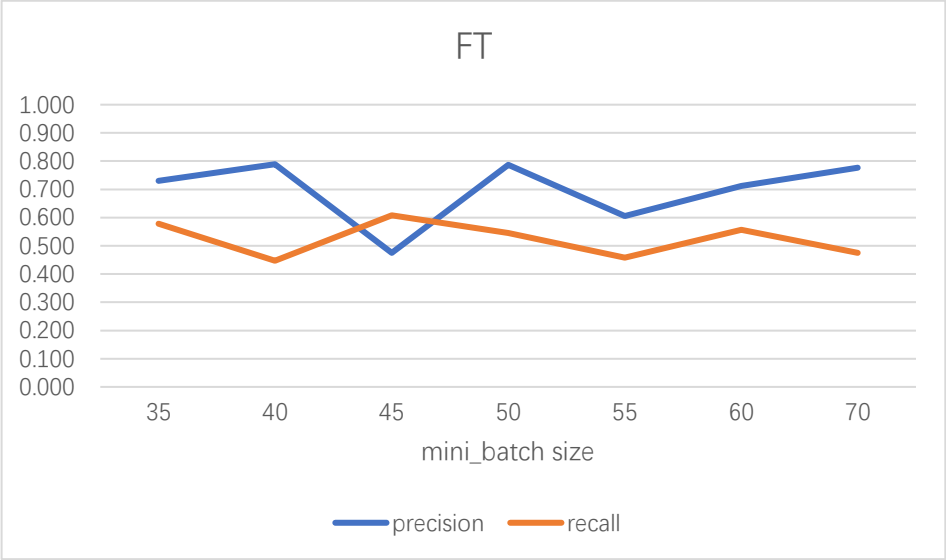

图 5-3　FT 类分类准确率与分类召回率

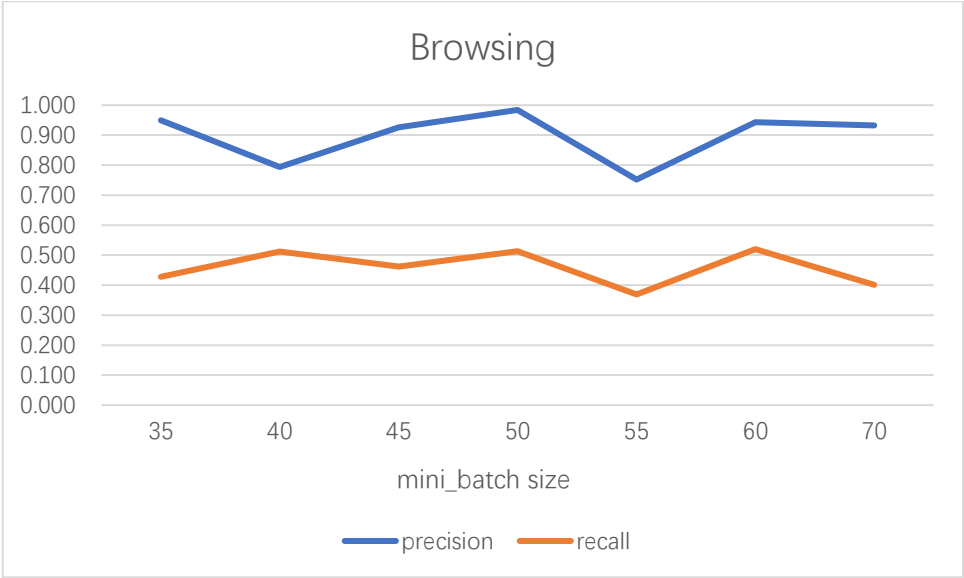

图 5-4　Browsing 类分类准确率与分类召回率

图 5-4 为基于深度神经网络加密流量分类模型对 Browsing 类的分类准确率和分类召回率随着 mini-batch size 的变化的统计图。在图中，Browsing 类的分类准确率始终远高于分类准确率，Browsing 类的分类准确率在 0.75~0.98 之间变化，而分类召回率则在 0.37~0.52 之间变化。当 mini-batch size 为 55 时，分类准确率取得最低值 0.75，分类召回率也取得最低值 0.37。

图 5-5 为基于深度神经网络加密流量分类模型对 Streaming 类的分类准确率与分类召回率随着 mini-batch size 变化的统计图。Streaming 类的分类准确率在 0.45~0.79 之间变化，分类召回率在 0.36~0.85 之间变化。当 mini-batch size 为 40 时，Streaming 类的分类准确率取得最大，为 0.79，分类召回率最低，为 0.36。

当 mini-batch size 为 50，分类准确率最低为 0.45，当 mini-batch size 为 70，分类召回率最大，为 0.85。

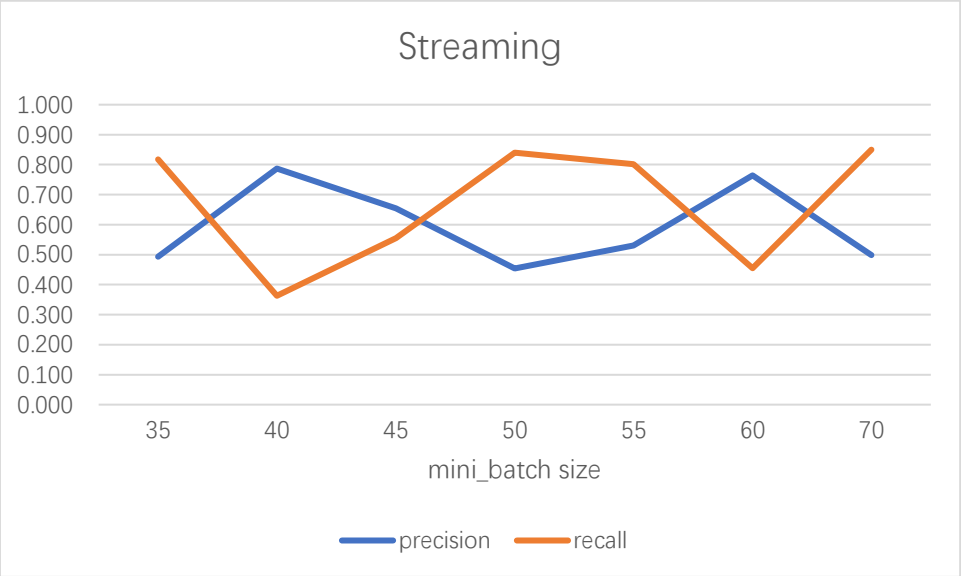

图 5-5　Streaming 类分类准确率与分类召回率

图 5-6 为基于深度神经网络加密流量分类模型对 P2P 类随着 mini-batch size 变化的分类准确率和分类召回率的统计图。在图中，分类准确率的变化范围为 0.40~0.86，分类召回率的变化范围为 0.58~0.82，可知分类准确率的变化范围比分类召回率大。当 mini-batch size 为 55 时，P2P 类的分类准确率最低，为 0.40，但 P2P 类分类召回率却最大，为 0.82。当 mini-batch size 为 60 时，P2P 类的分类准确率最大，为 0.86。当 mini-batch size 为 70 时，P2P 类的分类召回率最低为 0.58。

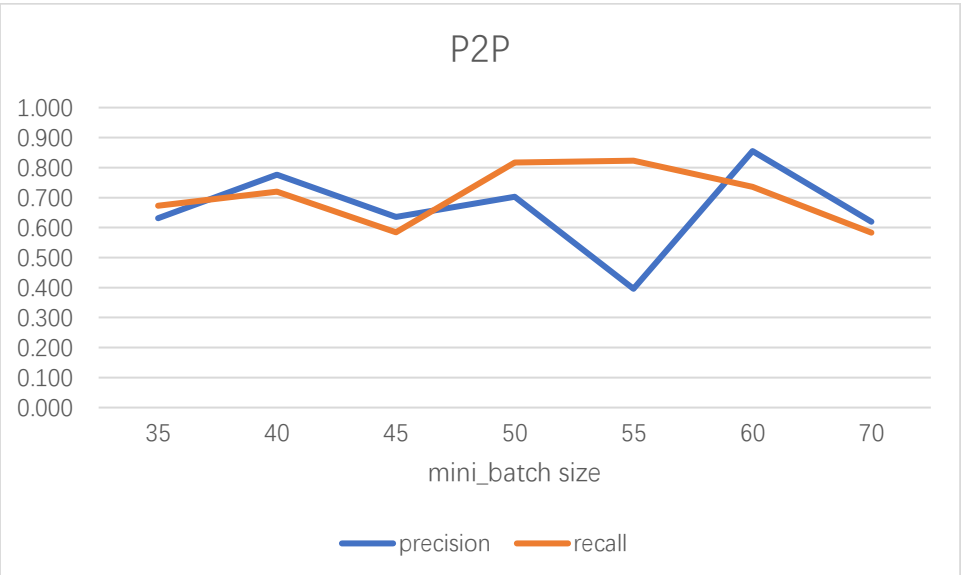

图 5-6　P2P 类分类准确率与分类召回率

图 5-7 为基于深度神经网络加密流量分类模型对 Email 类随着 mini-batch size 变化的分类准确率和分类召回率的统计图。随着 mini-batch size 的变化，Email 类

的分类准确率始终远大于分类召回率。在图中，Email 类的分类准确率在 0.72~0.97 之间变化，分类召回率在 0.15~0.19 之间变化，分类准确率远大于分类召回率，分类准确率的变化范围也比分类召回率大，分类召回率的波动极小。当 mini-batch size 为 50 时，分类准确率最低，为 0.72，当 mini-batch size 为 35 时，分类准确率最高，为 0.97。当 mini-batch size 为 55 时，分类召回率最低，为 0.15，当 mini-batch size 为 60，分类召回率最高，为 0.19。

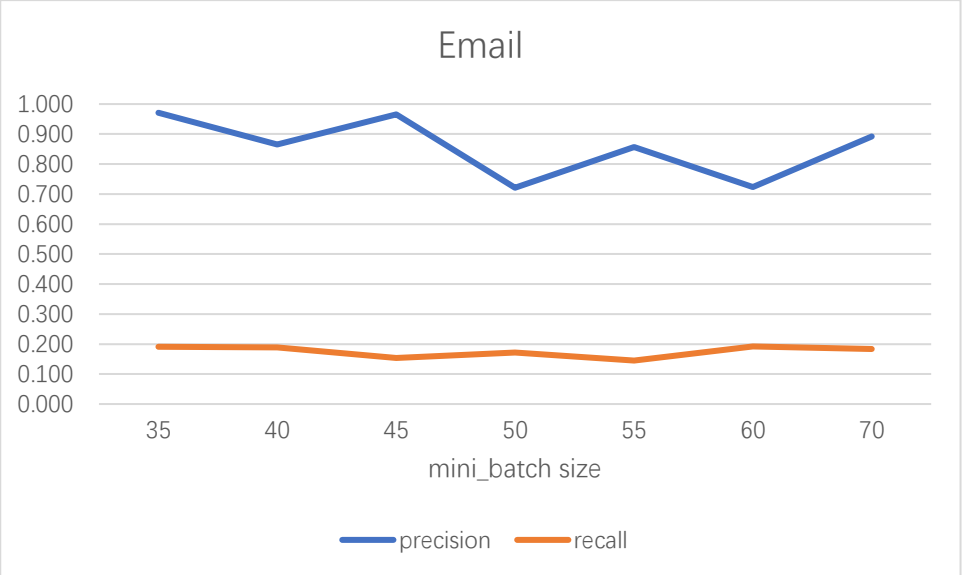

图 5-7　Email 类分类准确率与分类召回率

样本数量最多的 VoIP 类、Chat 类的分类准确率高于其余类别，受 mini-batch size 变化的影响也较小，最大差值分别为 0.168、0.126。而样本数量较少的 P2P 类的分类准确率受 mini-batch size 影响最大，最高为 0.855，最低为 0.396，最大差值为 0.459。由此可见，mini-batch size 改变会对深度神经网络分类模型产生不同程度的影响，各个类别受影响的程度与类别所占的样本数量比例有关。当 mini-batch size 为 40 时，分类模型总体的分类准确率最高，分类模型的分类能力最好。而当 mini-batch size 为 50 时，分类模型总体的分类召回率更高。

表 5-1 各个类别的平均分类准确率和平均分类召回率

流量类别	Ave-precision	Ave-recall
VoIP	0.815	0.925
Chat	0.845	0.789
FT	0.696	0.524
Browsing	0.897	0.458
Streaming	0.597	0.669
P2P	0.659	0.705
Email	0.856	0.175

表 5-1 为各个类别的平均分类准确率和平均分类召回率，可以看出，深度神经网络分类模型在 Browsing 类的分类准确率最高，而对 P2P 类的分类准确率最低。同时，深度神经网络分类模型在 VoIP 的分类召回率远高于其他类别。而 Email 的分类准确率虽高，但分类召回率远小于其他类别，深度神经网络分类模型对 Email 类的识别能力最差。

虽然各个类别的分类准确率随着 mini-batch size 的变化而变化，当 mini-batch size 不变时，深度神经网络分类模型对各个类别间的分类能力几乎保持不变。深度网络分类模型对 VoIP、Chat、Browsing、Email 类的分类准确率较高，而对 VOIP、Chat 类的分类查全率较高，结果表明，样本类所占比例仍然会影响模型的分类能力，且样本数量所占比例较高的样本，深度神经网络分类模型对该类别的分类预测能力较强。

5.3 与朴素贝叶斯算法对比

图 5-8，5-9 分别为当 mini-batch size 为 40 时，深度神经网络分类模型与朴素贝叶斯分类算法的分类准确率和分类召回率对比图。在图 5-8 中，除了 VOIP 类以外，其余各个类别在深度神经网络分类模型的分类准确率要远高于朴素贝叶斯算法，朴素贝叶斯分类算法对样本数量最多的 VOIP 分类能力好于深度神经网络分类模型，而深度神经网络分类模型对其他样本的分类能力远好于朴素贝叶斯分类算法。而在图 5-9 中，深度神经网络分类模型在 Browsing、Chat、FT、P2P、VOIP 类的查全率要好于朴素贝叶斯，而朴素贝叶斯分类算法对 Email、Streaming 类的查全率要好于深度神经网络分类模型。总体而言，深度神经网络模型的对加密网络流量的分类预测能力要好于朴素贝叶斯分类算法。

由于数据集为不均衡样本数据集，各类别样本数量差别较大，对实验结果产生了不可忽视的影响。实验结果显示，深度神经网络分类模型对样本数量较多的 VOIP、Chat 的分类预测能力较强。

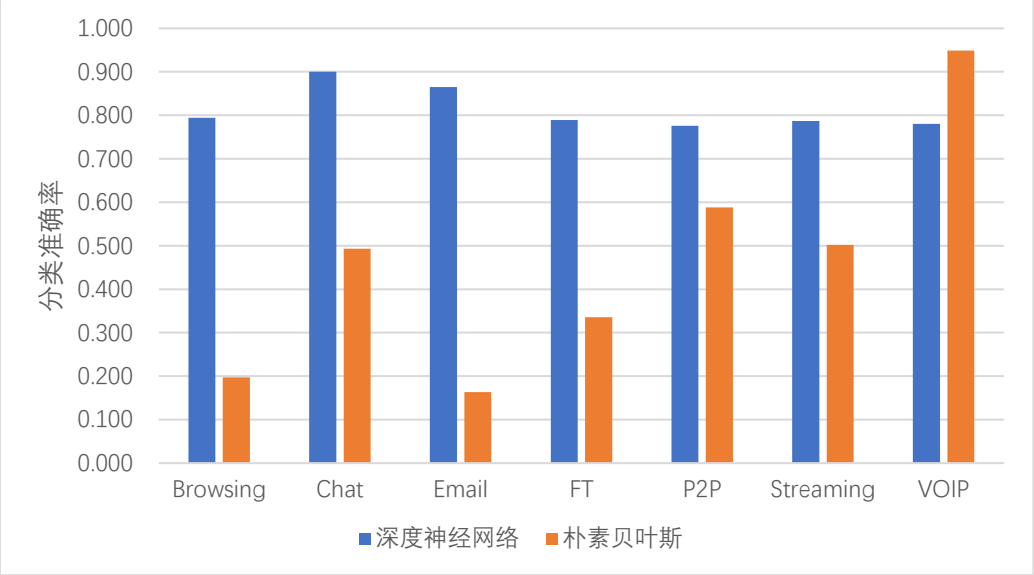

图 5-8 深度神经网络分类模型与朴素贝叶斯算法的分类准确率

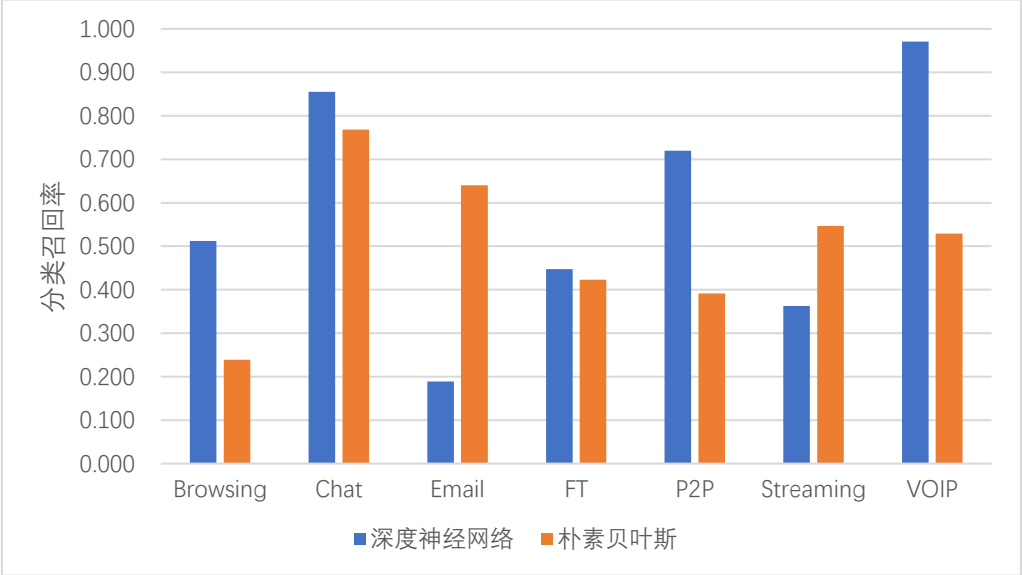

图 5-9 深度神经网络分类模型与朴素贝叶斯算法的分类召回率

第 6 章 总结与展望

6.1 总结

本文首先总结了传统的三种网络流量分类方法：基于端口匹配的分类方法，基于 DPI 的分类方法、基于协议解析的分类方法，并分析了这些传统方法的挑战不再适应当前复杂多变网络环境的原因。同时，本文分析了当前研究的前沿，基于统计学习的加密流量分类方法，通过运用网络流量的统计特征等信息，使用机器学习算法对网络流量分类，并提出了基于时序特征的深度神经网络加密网络流量分类方法。

本文提出基于时序特征的深度神经网络加密网络流量分类，通过统计网络流基于时序的一系列特征，同时运用了深度神经网络分类模型对其进行研究和分析。对经由 VPN 传输的 Browsing、Chat、Email、FT、P2P、Streaming、VOIP 等 7 种加密网络流量，通过提取加密网络流量基于时序的特征构成向量，转换成深度神经网络的输入，构造并训练深度神经网络分类模型，构建了基于深度神经网络的加密网络流量分类模型。

本文的主要工作如下：

1．分析现有的网络流量分类方法。

2．运用 ISCXFlowMeter 生成网络流，并提取经由 VPN 加密的网络流基于时序的特征。

3．设计并构建了基于深度神经网络的加密流量分类模型，对多种经由 VPN 加密的 HTTP、Email、P2P、Chat、Browsing、FT、VoIP 等 7 种网络流量进行分类，并分析影响分类模型对不同类别网络流量分类预测能力的原因。

4．在深度神经网络中，影响分类模型分类能力的因素众多，本文主要探讨了不同 mini-batch size 对基于深度神经网的加密流量分类模型分类能力的影响。在训练时，mini-batch size 会对深度神经网络模型产生不同程度的影响，当 mini-batch size 为 40 时，深度神经网络模型的分类能力最好。

5．本文将基于深度神经网络的加密流量分类模型与朴素贝叶斯分类算法对比，探究不同的分类算法对加密网络流量的分类能力。实验结果表明，基于深度神经网络分类模型的加密流量分类模型相比朴素贝叶斯算法能更好的识别加密网络流量。

6．在本文采用的网络流量数据集中，各个类别的网络流量样本数量差异明显，本文也探讨了多类样本数量不均衡数据集对模型分类能力的影响。

本文的创新之处在于通过提取加密网络流量基于时序的特征，构建了基于深度神经网络的加密流量分类模型，模型的分类能力远超于朴素贝叶斯算法，模型

的扩展性和可移植性较好。同时，本文探讨了在深度神经网络训练过程中，mini-batch size 的变化对深度神经网络模型训练的影响，以及对网络流量分类的影响。

本文构建了基于时序特征的深度神经网络加密网络流量分类模型，发现基于时序特征的深度神经网络的加密网络流量分类模型能有效分类加密流量。而 mini-batch size 的大小变化也会影响深度神经网络模型的分类能力，实验发现，当 mini-batch size 为 40 时，深度神经网络分类模型的分类能力最好。同时，将本文提出的基于时序特征的深度神经网络的加密流量分类模型与传统的朴素贝叶斯分类算法进行对比实验，结果显示，基于时序特征的深度神经网络加密网络流量分类模型的分类能力更好。然而，由于数据集各个类别的样本数目的不均衡，对分类结果产生了不可忽视的影响。

6.2 展望

本文提出的基于深度神经网络的加密流量分类模型取得了较好的分类效果，但仍存在一些尚待解决的问题。

1．本文采用的数据集为非平衡数据集，在数据集中不同类数据的数量差别较大，导致了不同类别流量的分类准确率差别较大。例如，VoIP 类的有 11008 个样本，而 Email 类仅有 569 个样本。如何解决非平衡数据对加密流量分类的影响，是未来研究的方向之一。

2. 由于时间和资源有限，本文在研究的 mini-batch size 对深度神经网络模型分类能力的影响是，mini-batch 的大小仅在 35~70 之间，研究范围较小。

3. 由于现有的网络流预处理工具较少，本文仅提取了加密网络流量基于时序的特征。未来可从更多的角度提取加密网络流量的特征。

参考文献

[1] 中国互联网协会.中国互联网发展报告 2019. http://cnnic.cn/gywm/xwzx/rdxw/ 20172017 7056 / 201902/t20190228_70643.htm

[2] 中国信息通信研究院.《互联网发展趋势报告(2017-2018)》.http://www.cac.gov.cn/2018-04/25/c_1122741920.htm.2017.

[3] Karagiannis T, Broido A, Faloutsos M. Transport layer identification of P2P traffic[C]//Proceedings of the 4th ACM SIGCOMM conference on Internet measurement. ACM, 2004: 121-134.

[4] 中国互联网协会、国家互联网应急中心.《中国互联网站发展状况及其安全报告(2018)》.http://www.isc.org.cn/zxzx/xhdt/listinfo-36071.html.2018.

[5] Lotfollahi M, Zade R S H, Siavoshani M J, et al. Deep Packet: A Novel Approach For Encrypted Traffic Classification Using Deep Learning[J]. 2017.

[6] Shi H, Li H, Zhang D, et al. Efficient and robust feature extraction and selection for traffic classification[J]. Computer Networks, 2017, 119:1-16.

[7] Yang L, Dong Y, Rana M S, et al. Fine-Grained Video Traffic Classification Based on QoE Values[J]. Wireless Personal Communications, 2018, 103(4):1-18.

[8] Roughan M, Sen S, Spatscheck O, et al. Class-of-service mapping for QoS: a statistical signature-based approach to IP traffic classification[C]//Proceedings of the 4th ACM SIGCOMM conference on Internet measurement. ACM, 2004: 135-148.

[9] Shafiq M, Yu X, 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.

[10] Erman J, Arlitt M, Mahanti A. Traffic classification using clustering algorithms[C]// Proceedings of the 2006 SIGCOMM workshop on Mining network data. ACM, 2006: 281-286.

[11] Sun G, Chen T, Su Y, et al. Internet traffic classification based on incremental support vector machines[J]. Mobile Networks and Applications, 2018, 23(4): 789-796.

[12] Internet Assigned Numbers Authority (IANA). https://tools.ietf.org/html/rfc6335, 2011.

[13] Sherry J, Lan C, Popa R A, et al. BlindBox:Deep Packet Inspection over Encrypted Traffic[C]//ACM Conference on Special Interest Group on Data Communication. ACM, 2015:213-226.

[14] Korczyński M, Duda A.Markov chain fingerprinting to classify encrypted traffic[C]// INFOCOM, 2014 Proceedings IEEE.IEEE, 2014:781-789.

[15] Li H, Hu C. MP-ROOM: Optimal matching on multiple pdus for fine-grained traffic identification[J]. IEEE Journal on Selected Areas in Communications, 2014, 32(10): 1881-1893.

[16] Sen S, Spatscheck O, Wang D. Accurate, scalable in-network identification of p2p traffic using application signatures[C]//Proceedings of the 13th international conference on World Wide Web. ACM, 2004: 512-521.

[17] Paxson V. Bro: a system for detecting network intruders in real-time[J]. Computer networks, 1999, 31(23-24): 2435-2463.

[18] Roesch M. Snort: Lightweight intrusion detection for networks[C]//Lisa. 1999, 99(1): 229-238.

[19] Sandvine Incorporated. http://www.sandvine.com/.2010.

[20] ipoque GmbH. http://ipoque.org. 2010.

[21] Yang Liu, Jinfu Chen, Peng Chang, Xiaochun Yun. A Novel Algorithm for Encrypted Traffic Classification based on Sliding Window of Flow's First N Packets[C].2017 2nd IEEE International Conference on Computational Intelligence and Applications (ICCIA) 8-11， Sept. 2017, Beijing, China, PP,463-470.

[22] Iliofotou M, Pappu P, Faloutsos M, et al. Network monitoring using traffic dispersion graphs (tdgs)[C]//Proceedings of the 7th ACM SIGCOMM conference on Internet measurement. ACM, 2007: 315-320.

[23] Bonfiglio D, Mellia M, Meo M, et al. Revealing skype traffic: when randomness plays with you[J]. ACM SIGCOMM Computer Communication Review, 2007, 37(4):37-48.

[24] Rossi D, Mellia M, Meo M. Understanding Skype signaling[J]. Computer Networks, 2009, 53(2):130-140.

[25] Baset S A, Schulzrinne H G. An Analysis of the Skype Peer-to-Peer Internet Telephony

Protocol[C]// Infocom IEEE International Conference on Computer Communications. 2007.
[26] Suh K , Figueiredo D R , Kurose J , et al. Characterizing and detecting skype-relayed traffic[C]// IEEE Infocom IEEE International Conference on Computer Communications. IEEE, 2006.
[27] Constantinou F. Identifying Known and Unknown Peer-to-Peer Traffic[C]// Proc. 5th IEEE International Symposium on Network Computing and Applications, 2006. IEEE, 2006.
[28] Sen S, Wang J. Analyzing Peer-To-Peer Traffic Across Large Networks[J]. IEEE/ACM Transactions on Networking, 2004, 12(2):219-232.
[29] Crotti M, Dusi M, Gringoli F, et al. Traffic classification through simple statistical fingerprinting[J]. ACM SIGCOMM Computer Communication Review, 2007, 37(1): 5-16.
[30] Esteves A F, In P R M, Pereira M. On-line detection of encrypted traffic generated by mesh-based peer-to-peer live streaming applications: The case of GoalBit[C]//2011 IEEE 10th International Symposium on Network Computing and Applications. IEEE, 2011: 223-228.
[31] Bermolen P, Mellia M, Meo M, et al. Abacus: Accurate behavioral classification of P2P-TV traffic[J]. Computer Networks, 2011, 55(6):1394-1411.
[32] Zhou L J, Li Z T , Liu B . P2P Traffic Identification by TCP Flow Analysis[C]// null. IEEE Computer Society, 2006.
[33] Dhamankar R, King R. Protocol identification via statistical analysis (PISA)[J]. White Paper, Tipping Point, 2007.
[34] Korczyński M, Duda A. Markov chain fingerprinting to classify encrypted traffic[C]//IEEE INFOCOM 2014-IEEE Conference on Computer Communications. IEEE, 2014: 781-789.
[35] Yang J, Yuan L, Dong C, et al. On characterizing peer-to-peer streaming traffic[J]. IEEE Journal on Selected Areas in Communications, 2013, 31(9): 175-188.
[36] Korczyński M, Duda A. Classifying service flows in the encrypted skype traffic[C]//2012 IEEE International Conference on Communications (ICC). IEEE, 2012: 1064-1068.
[37] Dusi M, Este A, Gringoli F, et al. Using GMM and SVM-based techniques for the classification of SSH-encrypted traffic[C]//2009 IEEE International Conference on Communications. IEEE, 2009: 1-6.
[38] Zeng Y, Gu H, Wei W, et al. Deep-Full-Range: a Deep Learning Based Network Encrypted Traffic Classification and Intrusion Detection Framework[J]. IEEE Access, 2019.
[39] Anderson B,Mcgrew D.Machine Learning for Encrypted Malware Traffic Classification: Accounting for Noisy Labels and Non-Stationarity[C]//The, ACM SIGKDD International Conference. ACM, 2017:1723-1732.
[40] Saber A, Belkacem F, Moncef A. Encrypted Network Traffic Identification: LDA-KNN Approach[C]//9 ème édition du colloque Tendances dans les Applications Mathématiques en Tunisie Algérie et Maroc. 2019: 25.
[41] Liu Z, Wang R, Japkowicz N, et al. Mobile app traffic flow feature extraction and selection for improving classification robustness[J]. Journal of Network and Computer Applications, 2019, 125: 190-208.
[42] Shekhawat A S, Di Troia F, Stamp M. Feature analysis of encrypted malicious traffic[J]. Expert Systems with Applications, 2019, 125: 130-141.
[43] Niu W, Zhuo Z, Zhang X, et al. A Heuristic Statistical Testing Based Approach for Encrypted Network Traffic Identification[J]. IEEE Transactions on Vehicular Technology, 2019.
[44] Sun G, Liang L, Chen T, et al. Network traffic classification based on transfer learning[J]. Computers & electrical engineering, 2018, 69: 920-927.
[45] Dong S, Li R. Traffic identification method based on multiple probabilistic neural network model[J]. Neural Computing and Applications, 2019, 31(2): 473-487.
[46] Cao J, Fang Z, Qu G, et al. An accurate traffic classification model based on support vector machines[J]. International Journal of Network Management, 2017, 27(1): e1962.
[47] Piskac P, Novotny J. Using of Time Characteristics in Data Flow for Traffic Classification[J]. 2017.
[48] Wang W, Zhu M, Zeng X, et al. Malware traffic classification using convolutional neural network for representation learning[C]// International Conference on Information Networking. IEEE, 2017.
[49] Shen M, Wei M, Zhu L, et al. Classification of Encrypted Traffic With Second-Order Markov Chains and Application Attribute Bigrams[J]. IEEE Transactions on Information Forensics & Security, 2017, 12(8):1830-1843.

[50] Zuleika Nascimento, Djamel Sadok, Stˆenio Fernandes and Judith Kelner.Multi-Objective Optimization of a Hybrid Model for Network Traffic Classification by combining Machine Learning Techniques[C]. 2014 International Joint Conference on Neural Networks (IJCNN), July 6-11, 2014, Beijing, China, PP,2116-2122.
[51] Wei Wang, Ming Zhu, Jinlin Wang, Xuewen Zeng, Zhongzhen Yang. End-to-end encrypted traffic classification with one-dimensional convolution neural networks. 2017 IEEE International Conference on Intelligence and Security Informatics (ISI), Beijing, China, 22-24 July 2017, PP,43-48.
[52] Giuseppe Aceto, Domenico Ciuonzo, Antonio Montieri, Antonio Pescapé. Mobile Encrypted Traffic Classification Using Deep Learning. 2018 Network Traffic Measurement and Analysis Conference (TMA) ,26-29 June 2018, Vienna, Austria,pp.
[53] Wubin Pan, Guang Cheng, Yongning Tang. WENC:HTTPS Encrypted Traffic Classification Using Weighted Ensemble Learning and Markov Chain, 2017 IEEE Trustcom/BigDataSE/ICESS, Sydney, NSW, Australia, 1-4 Aug. 2017, PP,1723-1732.
[54] Quinlan J R. Induction of decision trees [J]. Machine Learning,1986, 1(1): 81-106.
[55] Quinlan JR.C4.5: Program for machine learning [M]. California: Morgan Kaufmann, 1993.
[56] Cortes C, Vapnik V. Support-vector networks [J]. Machine Learning, 1995, 20: 273-297.
[57] Li X, Qi F, Xu D, et al. An internet traffic classification method based on semi-supervised support vector machine[C]//2011 IEEE International Conference on Communications (ICC). IEEE, 2011: 1-5.
[58] Bar-Yanai R, Langberg M, Peleg D, et al. Realtime classification for encrypted traffic[C]//International Symposium on Experimental Algorithms. Springer, Berlin, Heidelberg, 2010: 373-385.
[59] Tan X B, Su X Q, Qian Q M. The classification of SSH tunneled traffic using maximum likelihood classifier [C] //In 2011 International Conference on Electronics, Communications and Control (ICECC). 2011.
[60] Caicedo-Muñoz J A, Espino A L, Corrales J C, et al. QoS-Classifier for VPN and Non-VPN traffic based on time-related features[J]. Computer Networks, 2018, 144: 271-279.
[61] Lashkari A H, Draper-Gil G, Mamun M S I, et al. Characterization of Encrypted and VPN Traffic Using Time-Related Features[C]//The International Conference on Information Systems Security and Privacy. 2016:94–98.
[62] Shen M, Wei M, Zhu L, et al. Certificate-aware encrypted traffic classification using second-order markov chain[C]//2016 IEEE/ACM 24th International Symposium on Quality of Service (IWQoS). IEEE, 2016: 1-10.
[63] Fu Y, Xiong H, Lu X, et al. Service usage classification with encrypted internet traffic in mobile messaging apps[J]. IEEE Transactions on Mobile Computing, 2016, 15(11): 2851-2864.
[64] Yuan Z, Xue Y, van der Schaar M. Bitminer: Bits mining in internet traffic classification[C]//ACM SIGCOMM Computer Communication Review. ACM, 2015, 45(4): 93-94.
[65] Sun G L, Xue Y, Dong Y, et al. An novel hybrid method for effectively classifying encrypted traffic[C]//2010 IEEE Global Telecommunications Conference GLOBECOM 2010. IEEE, 2010:1-5.
[66] Bernaille L, Teixeira R. Early recognition of encrypted applications[C]//International Conference on Passive and Active Network Measurement. Springer, Berlin, Heidelberg, 2007: 165-175.
[67] Bonfiglio D, Mellia M, Meo M, et al. Revealing skype traffic: when randomness plays with you[C]//ACM SIGCOMM Computer Communication Review. ACM, 2007, 37(4): 37-48.
[68] Bernaille L, Teixeira R, Akodkenou I, et al. Traffic classification on the fly[J]. ACM SIGCOMM Computer Communication Review, 2006, 36(2): 23-26.
[69] Moore A W, Zuev D. Internet traffic classification using bayesian analysis techniques[C]//ACM SIGMETRICS Performance Evaluation Review. ACM, 2005, 33(1): 50-60.
[70] Saeed A, Kolberg M. Towards Optimizing WLANs Power Saving: Novel Context-Aware Network Traffic Classification Based on a Machine Learning Approach[J]. IEEE Access, 2019, 7: 3122-3135.
[71] Aceto G, Ciuonzo D, Montieri A, et al. Mobile Encrypted Traffic Classification Using Deep Learning: Experimental Evaluation, Lessons Learned, and Challenges[J]. IEEE Transactions

on Network and Service Management, 2019.
[72] Gómez S E, Hernández-Callejo L, Martínez B C, et al. Exploratory study on Class Imbalance and solutions for Network Traffic Classification[J]. Neurocomputing, 2019.
[73] Lim H K, Kim J B, Heo J S, et al. Packet-based Network Traffic Classification Using Deep Learning[C]//2019 International Conference on Artificial Intelligence in Information and Communication (ICAIIC). IEEE, 2019: 046-051.
[74] Sun H, Xiao Y, Wang J, et al. Common Knowledge Based and One-Shot Learning Enabled Multi-Task Traffic Classification[J]. IEEE Access, 2019.
[75] Rish I. An empirical study of the naive Bayes classifier[C]//IJCAI 2001 workshop on empirical methods in artificial intelligence. 2001, 3(22): 41-46.
[76] Hall M . A decision tree-based attribute weighting filter for naive Bayes[M]. es, 2007.
[77] Jiang L, Zhang L , Li C , et al. A Correlation-based Feature Weighting Filter for Naive Bayes[J]. IEEE Transactions on Knowledge and Data Engineering, 2018:1-1.
[78] Jiang L, Li C , Wang S , et al. Deep feature weighting for naive Bayes and its application to text classification[J]. Engineering Applications of Artificial Intelligence, 2016, 52(C):26-39.
[79] Moore A, Zuev D, Crogan M. Discriminators for use in flow-based classification[R]. 2013.
[80] Wagner D, Schneier B. Analysis of the SSL 3.0 protocol[C]//The Second USENIX Workshop on Electronic Commerce Proceedings. 1996, 1(1): 29-40.
[81] Dierks T, Rescorla E. The transport layer security (TLS) protocol version 1.2[R]. 2008.
[82] Canadian Institute for Cybersecurity, VPN-nonVPN dataset (ISCXVPN2016) https://www.unb.ca/cic/datasets/vpn.html.
[83] Tensorflow .https://www.tensorflow.org/
[84] Nair V, Hinton G E. Rectified linear units improve restricted boltzmann machines[C]// International Conference on International Conference on Machine Learning. Omnipress, 2010:807-814.
[85] Kingma D P, Ba J. Adam: A Method for Stochastic Optimization[J]. Computer Science, 2014.
[86] Goyal P, Dollár, Piotr, Girshick R , et al. Accurate, Large Minibatch SGD:Training ImageNet in 1 Hour[J]. 2017.

致谢

终于，三年的硕士研究生生涯即将画下句点。有幸能在环境优美、学术氛围浓厚的武汉大学度过了人生的七年，在这里，结识了来自五湖四海的同学们，感受了多种文化的碰撞，收获了很多难忘的人生记忆。

感谢我的妈妈，一直给我巨大的力量，默默关心我，在我浮躁时提醒我，在我受伤时照顾我，忍受我的任性。感谢我的姑姑，带给我人生更多的可能性，让我在面临人生的选择时更加从容。

感谢我的朋友们，翟小影、吴伟坚、李歆韵、张慧等，即使隔着遥远的地理距离，我们也一起分享快乐，分担痛苦。感谢宿管阿姨在我受伤时对我的关心与照顾。感谢我的室友们，实验室的各位同学，感谢二班的各位同学，包容我，关爱我。感谢于雅梦老师对我的照顾。

衷心感谢我的导师吴黎兵教授，感谢老师的耐心指导，让我能顺利完成毕业设计。在研究生的生涯中，吴老师渊博的专业知识和勤恳的生活态度深深感染了我，让我意识到自身的不足，督促我不断的进步。也感谢夏有华、胡建总师兄对论文的指导与帮助。

最后，感谢参加论文评审的各位专家与老师！