摘　　要

网络流量分析是网络管理员对网络进行有效管理、维护和保障的重要手段。通过对网络流量数据的分析，可以了解网络当前的运行情况，在网络资源配置、异常检测等方面提供数据参考。本文从深度学习的角度，研究网络流量分析，重点研究通过优化流量分析模型来提高分析性能。

网络流量分析相关的研究主要有网络流量预测和网络流量分类两个方向，以往的研究者通常只对其中一点进行研究。本文从这两个问题出发，结合深度学习技术，总结出基于深度学习的网络流量分析模型。该模型全面描述了网络流量分析的过程，为数据建模和最终的应用都提供了帮助。

在网络流量预测方面，使用长短时记忆模型来对流量数据进行预测。又充分利用网络流量数据自相似和长相关的特点，在长短时记忆模型的基础上结合人工神经网络。在国内外三份真实采集的数据集上进行实验，实验表明改进的长短时记忆模型可以有效的对网络流量数据进行时序数列预测。在网络流量的业务分类方面，本文通过将网络流数据可视化的方式，直观的表明在图像分类问题上效果卓越的深度学习技术可以应用于流量数据分类。考虑到网络流数据的长度是可变的，首次将支持可变长度输入的卷积神经网络应用于网络流量分析领域。在实验中构造业务多分类和异常检测二分类两种任务情景，使用深度网络结构进行分类学习，实验结果表现出在实用性方面优于以往方法。

关键词：网络流量分析，流量预测，流量业务分类，深度学习

Abstract

Network traffic analysis is an important means for network administrators to manage, maintain and protect the network effectively. From the analysis, they can monitor running status, helpful to optimize resource configuration and anomaly detection. This paper studies the network traffic analysis from the perspective of deep learning and focuses on improving the analysis performance by optimizing the traffic analysis model

At present, the network traffic analysis divides into two directions: network traffic prediction and network traffic classification. Formerly, researchers took attention to one aspect. This paper start from these two directions, propose network traffic analysis model based on deep learning. The analysis model describes the structure of network traffic analysis also avail data modeling and application.

In terms of network traffic prediction, this paper proposes to use the Long-Short Term Memory (LSTM) model to predict the network traffic data. This paper also make the most of network traffic data self-similar and long-term correlation, propose a neural network combined with LSTM and Artificial Neural Networks (ANN). Experimental results show that LSTM can predict the timing sequence forces model better than current methods in the reality data set collected at home and abroad. In terms of the business classification of network traffic, this paper visualizes network traffic data and visually shows that the deep learning technology that is effective in image classification can be applied to traffic data classification. Considering that the length of network flow data is variable, the convolution neural network that supports variable length input is applied to the field of network traffic analysis for the first time. In the experiment, we constructed two task scenarios of network flow business classification and anomaly detection, through deep network structure for data classification training and testing. Experimental results show that it is superior to the previous methods in usability.

Key word: Network Traffic Analysis, Traffic prediction, Traffic classification. Deep learning

目　录

1 绪论

1.1 课题背景和意义

当前，基于 TCP/IP 技术的互联网飞速发展：新的网络技术不断出现，网络基础设施的规模不断扩大，网络交互日益活跃。网络作为工作，生活和学习的重要工具，已经在交通、医疗、互联网服务、教育等多方面影响公众的生活，成为日常社会中越来越重要的组成部分。在互联网飞速发展的背后，愈加复杂的网络环境也给网络研究人员提出越来越多的问题。其中，一个非常重要的问题在于，作为网络的服务提供商或者网络的管理团队以及人员，如何通过对网络流信息获取后进行分析来了解、管理、检测、优化现有的网络环境[1][2]。

与高速发展的互联网相对比，研究者们对网络行为变化的了解非常落后。互联网与常见的自然系统相比，其最大的不同在于具有明显的社会特性，由于人是互联网系统中的重要组成部分，而人的行为又是不可完全预测的，所以互联网具有复杂、多变及动态等特点。如何着眼于宏观，从大量的数据中去把握互联网具有的统计学特性与性质正在引起研究者们的注意。另一方面，探索互联网用户行为与变化规律也是网络管理者或服务提供商进行改进服务、制定策略的重要依据。互联网的飞速发展向互联网的管理人员提出了更为复杂的问题，随着网络在企业乃至于国家安全中占据越来越重要的地位，网络中的恶意攻击也成为互联网发展过程中一个不可忽视的因素。面对隐藏在暗处的“黑客”，如何及时发现他们的入侵行为是应对网络攻击的第一步，这也是网络管理中的一项重大难题。与为用户提供丰富多彩的网络应用相比，互联网管理技术始终落后于应用的发展。通过对以往网络流量的研究，来改进管理方法，优化资源配置，满足用户的需求，尽可能减少网络上的恶意攻击者对网络正常服务的破坏等等都是亟需解决的重中之重。

面对如此复杂的网络环境，对于网络流量分析的研究就应运而生了。网络流量是所有网络行为的载体，它是记录和反映互联网发展的重要依据，几乎所有网络相关的活动都是与网络流量相联系。作为网络行为的重要组成部分，通过对网络流量数据的抓取和分析，可以间接掌握网络的行为。根据既定的网络协议，多种不相同的网络服务、网络行为都可以格式化为统一的网络流量格式，让网络管理者可以从更高的角度来了解互联网，管理互联网。通过对网络流量的统计，研究者了解到过去网络中用户的使用情况，可以实现业务统计、网络计费等功能。通过应用网络流量的预测结果，能更好的规划网络资源，保证网络的服务正常。根据对网络流量中恶意流量的识别，可以很好的保护正常业务不受影响。

1.2 流量分析的研究现状

本节从网络流量研究的三个方面说明国内外对网络流量分析的研究现状，包括从网络流量不同粒度的定义上、网络流量分析的应用和网络流量分析所使用的模型上来叙述。

1.2.1 网络流量的不同粒度分析

对于网络流量的特征分析问题，根据研究对象的不同，会从几个不同的流量粒度上展开。根据现有的研究资料，研究者们一般将网络流量分为三种级别[3]，分别是比特级别（Bit-level）、数据包级别（Packet-level）、网络流级别(Flow-level)。

(1). 比特级别的网络流量分析主要是关注网络流量在数量上的特征，比如网络中的传输速度，网络吞吐和网络负载等情况[4]。

(2). 数据包级别的流量分析，大多关注 IP 分组的丢包率、抖动、延迟等情况。

(3). 网络流级别数据的划分主要是根据网络传输的源地址和目的地址与应用协议，文献[5]给出的定义是，一个由源 IP 地址、目标 IP 地址、源端口、目标端口、应用协议组成的 5 元组。它可以反应出网络流的连接情况和到达过程与间隔，更重要的是网络流数据间接反映了网络应用的局部特征[6]。

上述三个级别的网络流量中，粒度从小到大逐步增加，在不同粒度的网络流量上会表现出不同的特征和规律。有文章[7]指出，在毫秒级的时间尺度上，影响网络流量特征最主要是网络协议；较粗粒度的网络流量（小时以上）特征受外界影响较多。

1.2.2 针对网络流量分析的应用

研究者们通过对网络流量的分析，最终的目的是满足现有业务发展的需要或者为互联网用户提供更优质的服务。在针对网络流量分析的各种应用中，比较受重视的问题有对正常流量的预测、对流量业务的识别统计、对异常流量的检测等。

网络流量的预测是管理、规划、优化、控制网络中各种硬件和软件资源的一个重要前提[8]。通过以往的研究发现，网络流量自身的统计特征决定了其自身的可预测性[9][10]。在拥塞控制算法中，研究者需要根据已经获取到的历史数据，对未来某一时刻或者一段时间内的流量做出预测，将预测所得到的值作为分配资源的依据，在流量的峰值到来之前有所准备。动态实时的流量预测，还可以应用于路由选择，当网络路由中某个节点拥有多条可选的路径时，通过各条路径上历史流量数据预测接下来一段时间可能的流量数据，作为路由选择的依据，以最大效率的完成网络传输，保障用户的正常使用需求。使用分层编码视频传输业务的机构还可以通过对服务器端的流量进行预测，来决定是否要传输增强层，在保证服务正常工作的前提下，提高用户体验。

在复杂的网络环境中对异常流量进行检测，也是网络管理中安全管理的重要组成部

分。在 Heady R[11][12][13]等人提出的“保护检测响应恢复模型”（Protection Detection Response Recovery, PDRR）中，由异常检测构成的入侵检测系统组成该模型的第二个环节。

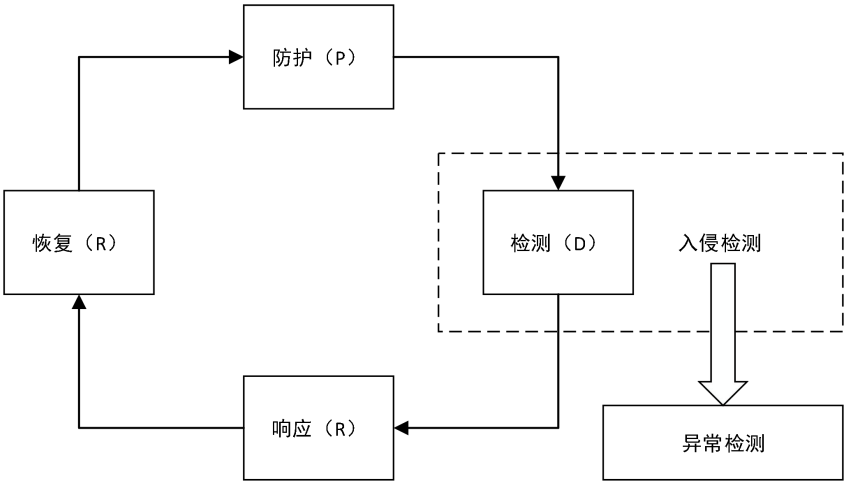


图 1.1 保护检测响应回复模型

1.2.3 用于网络流量分析的模型

在网络流量预测方面，随着计算科学领域的发展，时序数列预测领域已经有一些相对可靠的预测方法被提出去替代直观预测。早期 Nancy 等人[14]使用时间序列分析方法对 NSFNET 网络流量数据做了精细的预测，随之引发了学界一系列关于时间序列数据预测特别是网络流量数据预测问题的研究，建模方法包括自回归滑动平均模型（ARMA），自回归积分滑动平均模型（ARIMA）等[15][16]。大规模网络系统其本质是一个复杂的非线性系统，同时又会受到各种外界因素影响，它的宏观流量特征复杂多变。网络流量数据中既含有多种周期类波动，又会有非线性的上升、下降趋势，还有多种随机因素的干扰，使得用线性表述的流量模型表现出较大误差。所以，如何选择和优化非线性模型成为近年来预测网络流量的研究重点。其中 SVM[17]、LSSVM[18]、人工神经网络[19][20]、回声状态网络等都对预测精确度有一定的提高。但这些模型考虑到网络流量数据的非线性特征而忽视了流量数据的自相关性，所以如何将非线性时序数据预测与自相关特性联系在一起是一个值得探索的问题

在网络流量的业务分类方面，互联网发展早期的网络服务种类有限，网络中的业务类型数量相对而言还非常少，常见的有 Mail、Web、FTP、Telnet、DNS 等业务类型。这些数量有限的业务所使用的端口号会在因特网编号分配机构（IANA）进行注册，并使用唯一的固定端口号。最早的研究者们会基于特定的或者预先设定好的端口号来识别业务类型[21]。例如， Mail 业务使用的 SMTP 协议和 POP3 协议分别使用 25 端口和 110 端口；Telenet 业务使用的是 23 端口；DNS 服务使用的是 53 端口；Web 业务使用的 HTTP 协议使用的是 80 端口；FTP 使用的是 20 和 21 端口。基于端口号来识别网络流量业务

的方法在互联网的早期是有效的，只需要提取传输层的端口号就可以识别。网络管理人员在获取到网络流量数据后，分析数据中的端口号信息，并与常见的或者预先设定的端口业务信息进行匹配，匹配成功后就可以识别出该网络流属于哪一类网络业务类型。这种方法的最大优点在于方法简单，实现方便，所需要的信息极少，时间空间复杂度较低，可以快速高效率的完成网络流量业务分类。但是随着互联网的发展，互联网中业务类型不断增多，特别是各类 P2P 应用数量的增长，新的协议层出不穷，导致并不能保证所有的协议类型都在 IANA 中注册了其使用的端口。并且，很多服务器中应用所使用的端口会根据当前环境动态分配，甚至一些恶意软件会使用常用端口，伪装成为正常业务以骗过防火墙的封锁。如上所述，根据端口号识别网络业务类别的方法准确度逐渐下降，在对网络业务类型识别有较高要求的场景下失去作用。

随着根据端口识别网络流量业务类型的方法精确度下降，针对报文载荷内容的识别研究开始展开，基于报文签名的网络流量分类方法在 2002 年左右被提出。报文载荷的签名是报文中的一串字符或者若干位十六进制字符[22]。在基于报文载荷内容的识别方法中，需要网络管理人员在获取到网络流量数据之后对载荷数据进行提取，再根据上层协议的定义提取出待识别的应用协议在传输过程中出现的区别于其他协议的那么一串特征字段，这一串字符就是这个协议的特征串，即“协议签名”。在网络流分类过程中，首先使用深度报文检测技术（DPI）提取数据中每个报文的载荷部分，然后将载荷内容与预先定义的协议签名特征字段进行逐一查找匹配，若匹配成功则该网络流分类完成。例如：BitTorrent 协议中的特征字段为“0x13Bit”，MP2P 协议的特征字段是“GO”、“MD5”等。基于签名的网络流量业务识别方法同样实现起来非常简单，已经应用在很多协议识别系统上[23][24][25]。然而，当协议签名改变或者面对一种新产生的协议类型，人们必须手动去发现他们的特征签名，这一过程非常的耗费时间，致使这类方法实用性降低。并且由于在该识别方法中，需要逐个报文匹配已知的特征字段，时间复杂度相对较高。面对由于安全或者私密等原因，很多网络传输往往经过加密，该方法无法应对数据加密之后的分类问题也是缺点之一。由于以上种种原因，该方法只适用于小规模非加密的网络环境，不适用于日益复杂且多变的主干网络环境。

近几年来，基于特征提取和机器学习的自动分类技术在网络流量识别领域逐渐变得流行起来[26][27][28]。从统计的观点来看，常见的网络流量分类算法基于一种假设：常规流量分别具有某种相似的、异于其他流量的统计特征[29]。使用机器学习的网络流量分类技术基于网络流量中若干特征的提取，比如说数据包传输的间隔、数据包大小、端口号等等。将这些特征作为一些分类器的输入，例如朴素贝叶斯、决策树或者人工神经网络[30][31][32]。2005 年 Moore[33]等人从数据流中提取出 248 个统计特征（如流数据包平均达到时间等），并将朴素贝叶斯模型及其改进算法用于网络流分类之中，得到很好的分类

结果。这些方法的训练过程通常是离线并且耗时的，但在分类过程中效率较高，可以达到实时进行。该方法与上面的方法一样，需要网络专家先从网络数据中提取大量的特征信息，极度依赖专家经验。所以如何能让学习模型自动从网络流量数据中准确的提取有用的特征，是接下来研究的重点。

1.3 本文的研究内容

针对网络流量分析工作中存在的不足之处，本文重点针对网络流量分析所使用的模型展开研究。

本文的主要内容和创新之处：在概念上，从“量”和“质”的两个方面着手对网络流量数据进行研究，并将研究结果应用于正常流量的管理和异常流量的检测两个方面。其中在“量”方面主要研究网络流量比特级别的预测，在“质”方面主要研究数据流的业务分类。在应用上，正常流量管理可以是使用流量比特数的预测结果进行资源配置管理，也可以是根据网络流的分类结果进行统计和网络管理；异常流量的检测可以是“量”方面的异常，常见的有 DDoS 攻击，也可以是“质”方面的网络病毒攻击。

在模型的研究上，使用长短时记忆模型来预测网络流量，并考察了网络流量的自相关性。结合网络流量自相关的特点，将长短时记忆模型与人工神经网络相结合，进一步提升在粗粒度网络流量上的预测精确度。在网络流的业务分类方面，对以网络流为单位的流量数据进行预处理之后，将网络流量数据图形化，并在图形上应用卷积神经网络来分类。根据卷积神经网络可以有效自动提取图像特征的特性，在不同的数据集上设定不同分类情景进行试验，通过试验验证了深度学习优秀的特征提取能力，从而达到简化数据预处理过程的目的。针对 LeNet-5 结构的卷积神经网络只能输入大小长度固定的数据，然而网络流数据的长度是不等的，本文借鉴 Ye Zhang 设计的卷积神经网络，有效的解决数据长度不足末尾补零影响分类精确度的问题。在与其他分类模型的对比中，本文列举了使用卷积神经网络的众多优点，在识别加密协议、应对恶意攻击者伪装等问题上都相较其他模型有明显优势。

1.4 章节安排

根据论文的研究内容，本文一共分为五章，论文的章节安排如下：

第一章：绪论部分，这部分主要介绍了本文相关的研究背景，根据相关文献从网络流量分析的粒度、应用、模型等三个方面介绍了国内外研究现状，并简要介绍了本文的研究内容。

第二章：从网络流量分析的两个基本问题入手，并将应用归于正常流量管理和异常流量检测两个方面。结合深度学习的优势与特点，总结出一种基于深度学习的网络流量预测模型。

第三章：研究基于深度学习的网络流量预测算法。在递归神经网络的变种长短时记忆模型的基础上，结合网络流量自相关的特点，改进神经网络模型。并使用国内外的多份数据集上测试模型，将结果与递归神经网络、回声状态网络相比较，以验证模型的有效性。

第四章：研究基于深度学习的网络流量分类算法。将网络流量数据图形化之后的数据作为 LeNet-5 网络的输入，研究卷积神经网络对数据特征提取的有效性。针对网络流量长度不固定的问题，借鉴前人设计的卷积神经网络来解决这一问题。章尾对比国内外常用的几种流量分类方法，指出了本文模型在实用性方面的优势。

第五章：总结论文的主要贡献，并对下一步的工作做出展望。

2 基于深度学习的网络流量分析框架

2.1 网络流量分析解决的问题

面对日益复杂的网络业务以及逐渐增长的网络流量，通过互联网提供服务的互联网公司以及网络服务提供商都需要花费越来越多的工作来管理网络环境，保证网络正常、合理、高效运转。在这些工作中，绝大多数的业务都可以分为两类：对正常网络流量的分析以及对异常网络流量的检测。

2.1.1 正常流量分析

(1). 资源规划

根据历史网络流量数据，通过对网络流量分析系统中长期流量提供的合理分析，找到流量在未来会可能会变化的趋势。将人员、资金、设备等资源与业务流量进行有效的结合，合理分配现有资源，在保证业务正常开展、推广的前提下，提高资源利用率。对正常流量的分析可以为企业、机构在未来战略调整和资源分配上提供依据，最大限度的促进企业、机构发展。

(2). 应对热点业务的服务器扩容

很多通过互联网提供服务的产品，都需要应对热点业务的流量峰值。通常解决的传统方法有设备冗余和服务降级两种解决方案。设备冗余指在正常情况下，CPU 占用 30%以下，这会导致 CPU 资源不能充分利用以及无法应对四倍以上的流量峰值。服务降级是指在突发流量产生时，后端服务器系统将非核心业务以及周边业务进行降级，这会在一定程度上影响用户的使用体验。

如果想最大化设备利用率，并且保证用户使用体验，管理者面临的最大问题是扩容有一个繁琐的过程。将机房其他机器作为业务服务器将涉及配置管理数据库（CMDB）、上架装机、初始化、服务部署等多步。在互联网热点稍纵即逝的今天，很有可能导致当机器准备完毕，流量峰值也消失了的情形，即使随着云计算、云服务的发展，简化了扩容的步骤，面对日益火爆的互联网还是难以跟得上热点的步伐。

所以如何通过分析过去一段时间的流量数据，预测短时间内网络流量的多少是可以帮助企业通过及时的服务器扩容，来应对热点的到来。保证服务质量的情况下，提高服务器设备的利用率。

(3). 网络业务管控

随着网络的发展，互联网中的业务种类持续的增加，新兴的业务类型持续不断的出现。在早期互联网时代，网络流量主要由 HTTP、Email、SMTP、Telnet 等业务组成。随着互联网的飞速发展，传统业务所产生的流量所占据的比重持续减少，网络游戏、P2P、流媒体等业务流量在逐年上升，其中，P2P 和流媒体占据网络流量的大部分比重。P2P

服务平台的火热和在线视频内容提供商的发展，让这两类流量抢占了大部分的网络带宽，容易引起网络局部链路阻塞，给网络环境带来很多不好的影响。因此如果能通过分析网络流量数据，发现这类占据带宽较大的业务流量或者其他网络管理员不希望通过的流量，将有利于网络管理员进行网络的监控和管理。

2.1.2 异常流量检测

(1). 流量攻击预警

在众多网络攻击手段中，DDoS（Distributed Denial of Service）攻击是最常见也是最有效的攻击手段之一。DDoS 攻击使用众多的傀儡机，采用合法的请求不断的向目标服务器发送请求以达到占满服务器带宽，耗尽服务器资源的目的。由于 DDoS 攻击全部采用合法的请求手段，所以无法通过对流量的业务分类来判断是否受到攻击。那么通过对流量趋势的预测分析来实现流量攻击预警，以提前做好准备，以免被流量攻击压垮各类提供服务的软硬件。

(2). 网络病毒检测

计算机病毒是某些人通过程序代码对计算机资源进行破坏的恶意攻击手段，随着互联网的发展，针对网络资源的攻击已经成为近几年新兴计算机病毒的首选。现在的一些新式病毒已经不再简单依靠邮件附件来进行传播之后再在计算机内部启动运行，而是利用系统漏洞、网络端口来进行直接的恶意攻击。这样的攻击不需要“木马”在计算机内运行，所以传统的利用扫描文件来进行病毒检测的杀毒软件对此无能为力。如何将网络攻击流量与其他正常流量区分开，是网络流量分析研究领域面临的新挑战。

2.2 网络流量分析的基本问题

在网络环境中，网络流量数据产生于用户使用的终端设备与网络上其他终端设备进行交互。这些流量数据在底层硬件层上，以“0”、“1”二进制的形式进行传输。研究网络流量，可以从两个方面入手：从“量”的角度，本文研究网络传送的比特数，它与网络终端设备的网络吞吐量有关；从“质”的角度，本文研究二进制数据组成的数据帧，它与终端设备正在进行的业务有关。从这两个角度出发，总结得到网络流量分析的两个基本问题，那就是网络流量预测和网络流量分类。

2.2.1 流量预测

通过网络流量数据的采集，可以获得网络过去时刻的网络流量比特数，为了优化网络资源配置、及时发现网络流量过载攻击，需要对未来时刻的网络流量进行尽可能准确的预测。

在网络流量预测问题中，学习的任务是根据过去一段时间的历史流量数值，来预测未来某一时刻的网络流量。预测任务衡量的指标是预测的结果与真实流量值之间的偏差。学习的经验就是过去时刻网络流量抓包文件中获取到的流量比特数值。

2.2.2 流量分类

网络流量分析研究者利用网络数据抓包工具，获取到网络设备某一时间段完整的网络传输数据帧，每一个网络帧中封装了若干网络传输协议的数据，这些数据决定了网络终端是如何与其他终端进行数据传输以达到自己的目的的。比如，如果某一网络用户希望浏览网页，那么网络上传输的数据将包括：DNS 寻址以获取域名对应的 IP 地址，通过 IP 地址进行三次 TCP 握手以建立链接，使用 HTTP 协议传输网页内容，HTTP 传输结束后进行 TCP 四次握手以断开连接。这里的一切都是在遵循着固定的协议的情况下，将想要传送的数据转化为二进制发送的。为了能更好的管理网络，及时发现恶意流量，研究者需要对这些网络流量进行分类。

在网络流量分类问题中，研究人员需要学习的任务是根据当前网络终端上的网络数据判断业务类别，比如是一次网页浏览的业务还是一次某种网络病毒的攻击。衡量任务的度量是分类结果与真实业务类别之间是否存在偏差，常用的标准是查准率，查全率等。任务所依赖的经验来自于已知流量业务类别的网络流量历史数据。

2.3 深度学习概述

作为一种近几年火热的技术，深度学习（Deep Learning, DL）已经应用于字符识别、语音识别、图像理解、机器翻译等各个方面，并取得了巨大的成功。深度学习的概念是由人工神经网络（Artificial Neural Network, ANN）发展而来的，是指一类对具有深度结构的神经网络进行有效训练的方法。神经网络是一种由很多非线性神经元组成的分层系统，通常神经网络的深度不包括输入层的深度。从理论上来说，一个浅层结构的神经网络虽然在节点数足够的大时，也可能充分逼近地表达任意多元非线性函数，但这种浅层结构在实际应用中，往往需要大量节点而无法实际使用。通常来说，对于给定数量的训练样本，如果缺乏其他先验知识，大家更希望使用较少的计算单元来建立目标函数的表达，以获得更好的泛化能力。然而在网络深度不够时，这种结构表达方式可能无法建立。

最早期的神经网络是 McCulloch 和 Pitts 在 1943 年建立的 MP 模型，相关的线性回归方法甚至可以追溯到 1800 年左右。MP 模型是单个神经元的形式化数学表达，可以执行逻辑运算，然而不能够进行学习。1949 年，Hebb 对生物神经网络提出了“学习”的思想。1958 年，Rosenblatt 提出了感知器（Perceptron）模型与其相对应的有效学习算法。在随后的很长一段时间内，经历了神经网络研究的低潮期，但还是有很多研究者提出了许多神经网络的新模型。到了二十世纪八九十年代，这些新模型催生了神经网络的诞生，

并且掀起了新的一轮神经网络研究的高潮。其中比较有名的模型有：多层感知机、Hopfield 神经网络、玻尔兹曼机等。其中的多层感知机就是一个由多层节点构成的前馈神经网络（Feedforward Neural Network，FNN），其中每个非输入层的节点都具有非线性的激活函数，每一层与下面一层的神经元是全连接的。其中，Fukushima 提出的神经认知机是第一个拥有深度特征的神经网络。更关键的是，神经认知机促成了卷积神经网络模型的诞生。卷积神经网络在近几年的大规模数据比赛中，屡次打破纪录，获得了极大的成功。

在神经网络的训练过程中，反向传播方法（Backpropagation）是其中最常见也是最有效的方法，该方法最早由 Werbos 提出，并经过 LeCun，Parker 等研究者的完善而得到了极好的发展。但是，直到二十世纪八十年代，反向传播还只是对浅层神经网络有效。直到 1991 年，这个问题才作为深度学习的一个基本问题而得到理解。Hochreiter 指出，典型的深层神经网络存在梯度得消失和爆炸（Gradient Vanish, Gradient Exploding）。积累反向传播的误差信号在神经网络层数增长的时候，会出现指数衰减或者指数增长的现象，从而导致计算的数值迅速缩小为零或者超过数值边界。这个研究指出了为什么反向传播方法在深层网络训练时效果不佳的原因。同样，该问题在循环神经网络中也会出现。随着 1995 年之后支持向量机的迅速发展，神经网络的研究又进入低潮期。为了克服梯度消失或者爆炸的影响，在上个世纪末期，Hochreiter 对于该问题的思考推动了很多新方法的探索，其中比较重要的有：长短时记忆模型（Long Short-Term Memory Network）、基于 GPU 的计算机、海森无关优化(Hessian-Free Optimization)、权值矩阵空间的替代搜索。直到 2006 年，卷积神经网络以外其他多层神经网络模型的训练问题才受到学者的关注。

2006 年，Hinton 等人发表了两篇重要的论文，分别为《Neural Computation》上的“A fast learning algorithm for deep belief nets”和发表在《Science》上的“Reducing the dimensionality of data with neural networks”，标志着深度学习的开始。Hinton 说明了浅层神经网络的无监督学习有助于深层神经网络的有监督学习算法（比如反向传播算法）。特别是，其验证了通过无监督预训练和有监督调优构成的两步策略，不仅有效的帮助多层神经网络克服反向传播算法的失效问题，还使得深层神经网络具备更加优秀的学习能力。紧接着，更多的技术和方法被提出来，如最大池化、丢失链接、dropout 等，并且在很多问题上取得了大大优于传统方法的效果。这些新的技术和其获得的优异效果，使得深度学习技术受到业界的欢迎，从而发展成为神经网络的一次新浪潮。

2.4 基于深度学习的网络流量分析模型

深度学习技术可以通过复杂的神经元，寻找到训练数据中人为不易发现的特征，并将其应用到测试数据上，得到预测或者分类结果。并且，深度学习模型还可以随着经验的累积自动提高自己进行判断的准确率。根据上文的网络流量分析两大基本问题，将深度学习应用在其中，得到基于深度学习的网络流量分析模型如下图：

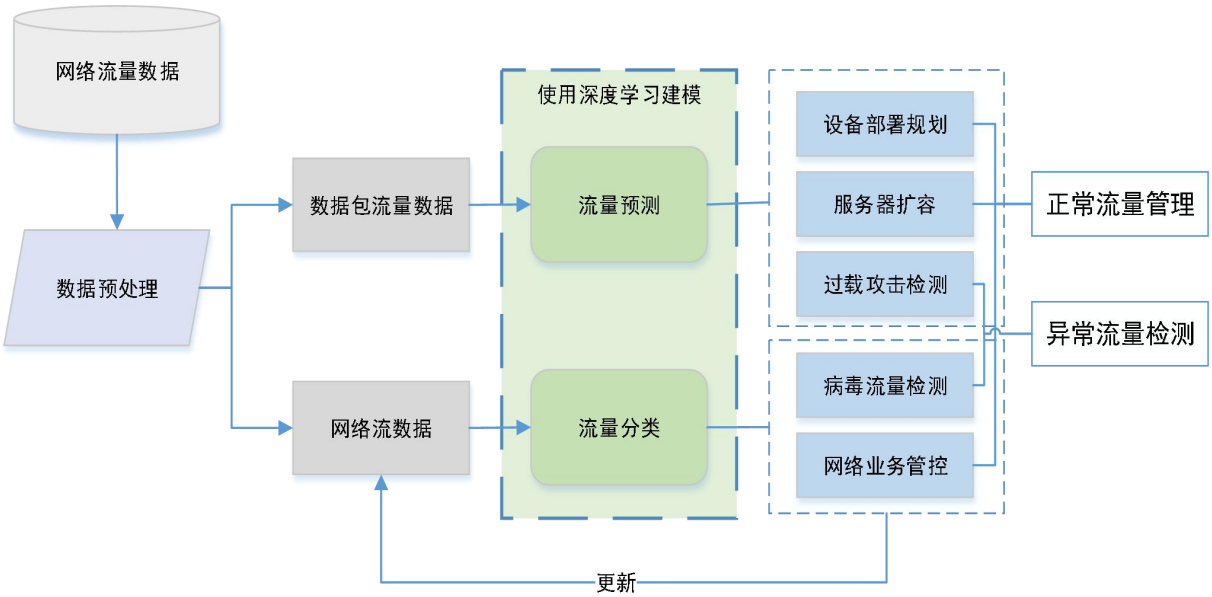


图 2.1 基于深度学习的网络流量分析模型

首先在网络环境中使用流量嗅探工具采集网络流量原始数据，根据需要进行预处理以后，得到本文需要分析的数据包流量数据和网络流数据，这两份数据分别代表了网络流量上的“量”和“质”的特征。在流量预测的任务中，需要使用能够反应时序特征的深度模型并结合网络流量特有的性质来进行网络流量预测。在流量分类的任务中，通过对带有标签的流量数据进行学习，利用深度学习模型强大的提取特征的优越性，来实现流量的分类。为了达到病毒流量检测的目的，模型训练数据中必须要包含多种一定数量的带标签的恶意攻击流量。通过对两类问题的学习结果，来辅助网络管理员或者决策者进行管理与决策。根据深度学习的特性，可以使用测试数据最终的实际标签来更新训练数据集，不断提升网络分析模型的准确度和实用性。

该模型融合了网络流量分析问题中需要解决的两类问题，从概念的角度指导网络流量分析。例如，网络异常检测系统的开发者可以根据该模型，从网络流量过载和异常流量检测两个角度出发构建系统。

2.5 本章小结

本章从概念上描述了网络流量分析系统中的两个基本问题和基于网络流量分析可以实现的应用。又简单叙述了深度学习的发展历史与深度学习的特点与优势。最后将这两者相结合，构建基于深度学习的网络流量分析模型，根据此模型指导后文的相关工作。

3　基于深度学习的网络流量预测研究

3.1 引言

TCP/IP 网络在现代社会中占据越来越重要的地位，如何更好的理解并且正确预测网络的行为成为信息技术发展中至关重要的一环。对于中/大型网络提供商来说，TCP/IP 网络预测已经成为一项重要的任务，并且获得越来越多的关注[34]。通过提升这项任务的准确度，网络提供商可以更好的优化资源，提供更好的服务质量[35]。不仅如此，网络流量预测可以帮助检测网络中的恶意攻击。例如拒绝服务或者垃圾邮件攻击可以通过比较真实流量和预测流量被检测出[36]。越早检测出这些问题，就可以获得越可靠的网络服务。

3.2 网络流量的性质

针对网络流量预测问题，首先应该充分了解网络流量的特性，才能建立尽可能简单而又准确的网络流量预测机器学习模型。本节介绍网络流量的自相似性和长相关性两个重要的性质。

3.2.1 自相似性

自相似性起源于混沌领域，最初用于表述一些形状特点与观察尺度大小没有关系的样本，例如地貌、山脉等等。伴随着研究的深入，研究者渐渐认识到在自然界中，自相似性是一种非常普遍的现象。在自然界中，云朵、闪电、雪花的图案在不同尺度大小下面观察，都是相似或者相同的。

直观的理解自相似性的概念，可以按照字面理解为：某事物与自身在外形或者特性上的相似性。如将一个几何体分为若干份，每一份都与这个整体存在相似或者形状保持一致的关系。举例来说，图 3.1 体现了这一种特性——在三角形 a 上，将其按照一定的规则划分成为 b、c、d 的形状，然后将三角形 a 缩小后可以在三角形 b、c、d 中找到与 a 形状相同的部分。换个角度来看，将三角形 d 的部分区域逐渐放大，可以看到形状与 a 一致的图形。自相似性的严谨表述方式是：在不同的观测尺度下，图形的局部特征与图形的整体特征是相似甚至是相同的，这部分的区域具备图形整体的形态、规则性、复杂程度等特性。图 3.1 所举的例子是相对严格的一类自相似的现象，称之为：确定自相似（Deterministic Self-similar）；在社会与自然中，更常见的一种自相似性是自相似性中的随机自相似（Stochastic Self-similar）。在本文中所涉及到的自相似性都是指的随机自相似性。

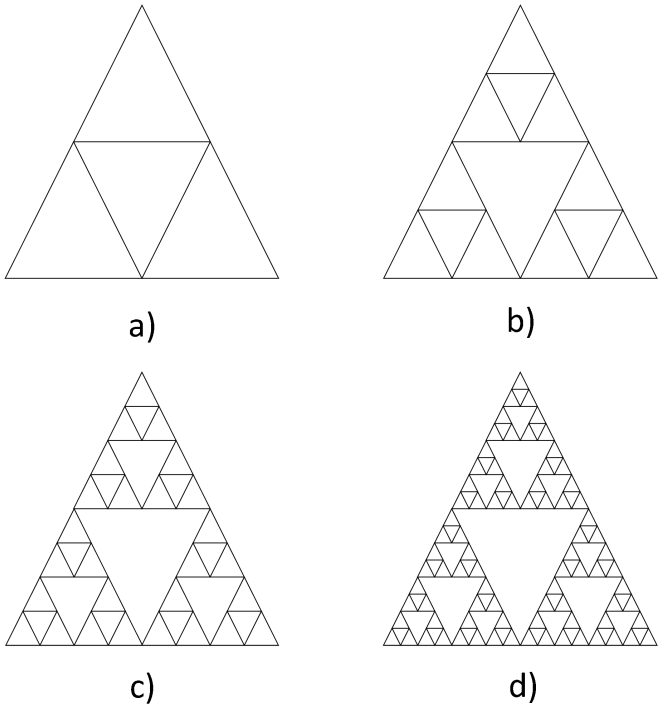


图 3.1 自相似图形示例

统计学中的自相似性指的是随机过程中的时间序列在时间维度上表现出自相似的性质，即这个时间序列在不同时间尺度下观察都表现出相同或者相似的的“规律”，这里的“规律”是这个随机过程在时间序列上隐藏的某些统计特性。从分布的角度来定义随机过程中的自相似性：令 t 代表时间，对于所有 $a>0$ ，一个随机过程 $\{Z(t)\}$，如果 $\{Z(at)\}$ 与 $\{a^H Z(t)\}$ 具有相同的有限维分布，即：

$$\{Z(at_1),Z(at_2),...Z(at_n)\} =_D \{a^H Z(t_1),a^H(t_2),...a^H(t_n)\} \tag{3.1}$$

那么 $\{Z(t)\}$ 是自相似随机过程。其中 n 代表 n 维分布；参数 H 称为 *Hurst* 参数，它是表述统计过程中自相似性的一个指标，H 的取值范围为 $0.5<H<1$，H 取值越大，自相似程度越高；D 代表英文单词“Distribution”——分布，即“ $=_D$ ”符号两边有相同的分布。事实上，Hurst 参数的取值范围是：$0<H<1$，当 $H<0.5$ 时，随机过程呈现为一种“随机”状态，即不同时间的状态转换是相互无关的；当 $0<H<0.5$ 时，表明随机过程是某种短程相关过程，也就是不具有自相似性质；当 $0.5<H<1$ 时，随机过程具有自相似性，也就是一种长程相关过程。

在上述公式（3.1）的定义中，条件过于严苛，因此又有基于随机过程二阶矩的二阶自相似性数学定义。如果一个平稳随机过程 $\{X(t)\}$ 的自相关函数 $r(k)$ 满足公式（3.2）：

$$r(k)=\frac{1}{2}[(k+1)^{2H}-2k^{2H}+(k-1)^{2H}] \tag{3.2}$$

则*{X(t)}*被称为严格二阶自相似过程。其中，*k* 代表自相关函数中的时间间隔；*H* 依然是 *Hurst* 参数。严格二阶自相似过程的条件依旧比较严苛，所以给出渐近二阶自相似过程的定义——如果平稳随机过程*{X(t)}*的自相关函数满足公式（3.3）：

$$\lim_{m\to+\infty} r^{(m)}(k)=\frac{1}{2}[(k+1)^{2H}-2k^{2H}+(k-1)^{2H}] \tag{3.3}$$

则这个随机过程被称为渐近二阶自相似过程。其中 $r^{(m)}$ 代表*{X(t)}*的 *m* 阶聚集序列 $X^{(m)}=\{X^{(m)}{}_i\}$ 的自相关函数， $X^{(m)}$ 中的第 *i* 个元素 $X_i^{(m)}$ 定义如下：

$$X_i^{(m)}=\frac{1}{m}\sum_{t=m(i-1)+1}^{mi} X(t) \tag{3.4}$$

这里的 *m* 和 *i* 都取正整数，*{X(t)}*为离散时间序列或者是经过离散化处理后的时间序列。对于平稳离散时间随机过程*{X(m)}*，它的自相关函数 *r(k)*如公式所示：

$$r(k)=E[X(m)X(m+k)] \tag{3.5}$$

在公式（3.5）中，平稳随机过程*{X(m)}*的期望必须为零。在真实世界采集的数据应用于理论研究与工程应用时，经常遇到随机过程期望不为零的情况。当研究期望不为零的随机过程自相似性时，应该采用自协方差函数 *c(k)*来消除期望不为零的影响：

$$c(k)=E[(X(m)-M_m)(X(m+k)-M_{m+k})] \tag{3.6}$$

其中， M_m 和 M_{m+k} 分别是*{X(m)}*和*{X(m+k)}*的期望。根据自协方差函数与自相关函数的关系可得：

$$c(k)=E[(X(m)-M_m)(X(m+k)-M_{m+k})]=r(k)-M_mM_{m+k} \tag{3.7}$$

当*{X(m)}*和*{X(m+k)}*的期望为零时，M_mM_{m+k} 等于零，则 *r(k)*与 *c(k)*相等。

3.2.2 长相关性

如果一个时间序列的随机过程在不同时间尺度下或者在长时间范围内都表现出明显的相关性，即一个时间序列随机过程的自相关函数满足如下条件，则该随机过程具备长相关性：

$$\sum_{k=1}^{+\infty}|\mathrm{r(k)}|=+\infty \tag{3.8}$$

长相关性和自相似性在概念上有一定的联系与区别。长相关性与自相似性是不同领域中的定义，然而在研究网络流量分析过程中，二者又相互关联。自相似性是描述网络流量在不同时间尺度下的性质，当时间尺度确定时，网络流量的时序数列表现为长时间跨度上的长相关性。长相关性反映了随机过程自相似性中的持续现象，是自相似性的一个重要特征。

然而，长相关性是一种渐近的统计特性，用于描述时间序列随机过程的自相关函数

在大时间延迟下的变化规律，长相关性在固定有限值延迟下不具备统计意义。对于离散时间序列来说，如果它的自相关函数 *r(k)*在时间延迟 *k* 逐渐趋向于正无穷的变化过程中衰减得很慢，如双曲函数衰减或者更慢，那么这个平稳的离散时间序列是具有长相关性的；与之相反的，如果 *r(k)*衰减得很快，如呈指数衰减或者更快，那么这个时间序列不具有长相关性，是短相关的时间序列。

显然，随着时间延迟 *k* 的逐渐增大，指数函数衰减的速度快于双曲函数。根据数学的相关知识：呈双曲函数衰减（或更慢）的自相关函数不可加。上述公式（3.8）指出了自相关函数是绝对不收敛的。某些独立的短相关随机过程在 $k \neq 0$ 的情况下，其自相关函数均为零，另外一些短相关随机过程的自相关函数等价于指数函数的倒数或者等价于高次幂函数的倒数，如公式（3.9）（3.10）所示：

$$r(k) \sim k^{\alpha}, \alpha < -1 \tag{3.9}$$

$$r(k) \sim \beta^{-k}, \beta > 1 \tag{3.10}$$

即 *r(k)*图形的随时间减小的速度要快于 k^{-1} 的速度，呈现指数衰减，因此它们的自相关函数收敛，即具有可加性：

$$\sum_{k=1}^{\infty} | r(k) | < +\infty \tag{3.11}$$

基于上述原因，长相关的随机过程应该具备 *r(k)*的非可加性，也就是要自相关系数呈双曲衰减或者更慢。

3.3 基于深度学习的流量预测模型

本节介绍用于网络流量预测的递归神经网络(Recurrent Neural Networks , RNN)，以及 RNN 的一种特殊类型——长短时记忆网络(Long Short Term Memory networks, LSTM)。在此基础上，结合网络流量自相关性的特点，总结出适用于自相关序列的改进长短时记忆模型。

3.3.1 递归神经网络

递归神经网络是近年来机器学习与深度学习领域比较热门的学习方法，与传统的前馈神经网络(Feedforward Neural Network, FNN)不同之处在于，FNN 的神经元通过输入层、隐藏层、输出层的连接进行信息的传递，各个输入、输出项之间相互独立，同一层的神经元之间往往没有连接。而 RNN 在网络中引入了隐藏层循环的结构，建立了隐藏层神经元到隐藏层神经元自身的连接。通过这种环状结构，神经元可以将上一时刻的输

入的信息“记忆”在神经网络中，并对当前时刻的输出产生影响。所以 RNN 更能良好的反应数据的在时间上的先后关系，在时序数据的预测问题上，往往有着比 FNN 更优异的表现。

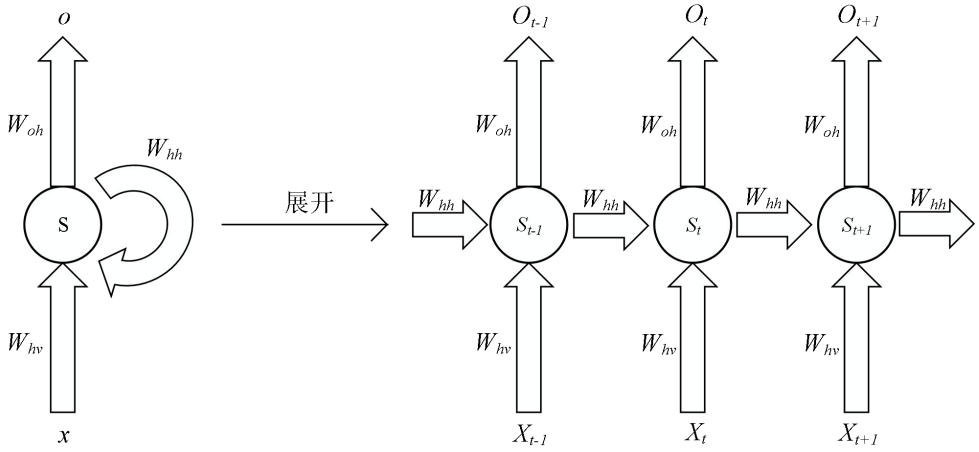


图 3.2 递归神经网络结构图

使用 RNN 模型的预测过程和 FNN 类似，都使用前向传播算法来计算：

$$
\begin{aligned}
&For\ t\ \ from\ 1\ to\ T\ \ do\\
&\quad u_t \leftarrow W_{hv}v_t + W_{hh}h_{t-1} + b_n\\
&\quad h_t \leftarrow e(u_t)\\
&\quad o_t \leftarrow W_{oh}h_t + b_o\\
&\quad z_t \leftarrow g(o_t)\\
&End\ \ for
\end{aligned}
\tag{3.12}
$$

其中 T 为输入数据的长度，v_t 为 t 时刻的输入，z_t 为 t 时刻的输出，W_{hv}、W_{hh}、W_{oh} 分别为输入层到隐藏层、隐藏层到隐藏层、隐藏层到输出层的连接矩阵。e 和 g 为隐藏层和输出层的激励函数。

RNN 的训练过程与 FNN 有一定的区别，FNN 通常通过后向传播算法(Back Propagation, BP)来训练神经网络，而 RNN 因为前面若干时刻的隐藏层状态也会影响输出层产生的误差，所以需要沿着时间维度向前对后向传播的结果误差进行叠加，即时间后向传播算法(Back Propagation Through Time, BPTT)。RNN 的时间后向传播算法首先定义损失函数对神经元 j 在时刻 t 输入值的偏导，然后应用链式求导法则来计算损失函数对网络权重的偏导：

$$
\begin{aligned}
&For\ t\ \ from\ T\ \ down\ to\ 1\ do\\
&\quad do_t \leftarrow g'(o_t)\cdot\frac{dL(z_t,y_t)}{dz_t}\\
&\quad db_o \leftarrow db_o + do_t\\
&\quad dW_{oh} \leftarrow dW_{oh} + do_t h^T{}_t\\
&\quad du_t \leftarrow e'(u_t)\cdot dh_t\\
&\quad dW_{hv} \leftarrow dW_{hv} + du_t v^T{}_t\\
&\quad db_h \leftarrow db_h + du_t\\
&\quad dW_{hh} \leftarrow dW_{hh} + du_t h^T{}_{t-1}\\
&\quad dh_{t-1} \leftarrow W^T{}_{hh} du_t\\
&End\ \ for\\
&\mathrm{Re}turn\ d\theta = [dW_{hv}, dW_{hh}, dW_{oh}, db_h, db_o, dh_0]
\end{aligned}
\tag{3.13}
$$

损失函数与神经元之间的偏导数由当前时间 *t* 的输出层与下一时刻 *t+1* 隐藏层的影响。对每个时间点上利用链式求导法则，将所有的结果在时间维度上进行相加，得到损失函数对于神经网络权重 *w* 的偏导数。最后使用梯度下降法(Gradient Descent)，更新递归神经网络中的权重，直到满足条件。

在 RNN 训练过程的最后一步可以看到，梯度在反向传播的过程中，每一步都要与 W_{hh}^T 相乘。如果特征值 $W_{hh}>1$，这将导致梯度爆炸(Gradient Explode)；如果特征值 $W_{hh}<1$，这将导致梯度消失(Gradient Vanish)[37][38]。针对这个问题，Hochreiter 提出了长短期记忆(Long Short Term Memory, LSTM)神经网络[39]。

3.3.2 长短周期递归神经网络

长短周期记忆模型是一种特殊的 RNN 模型，其关键在于将 RNN 中隐含层的神经元替换成为细胞状态。细胞状态在时间链上相互传递，只有一些少量的线性交互，所以细胞单元上的信息很容易保持。每个记忆体中包含一到多个记忆细胞(Memory Cell）和三种非线性求和单元。非线性求和单元又被称作“门”(Gate)，分为 3 种：“输入门”(Input Gate）“输出门”(Output Gate）和“遗忘门”(Forget Gate)，分别通过矩阵乘法控制记忆细胞的输入、输出。

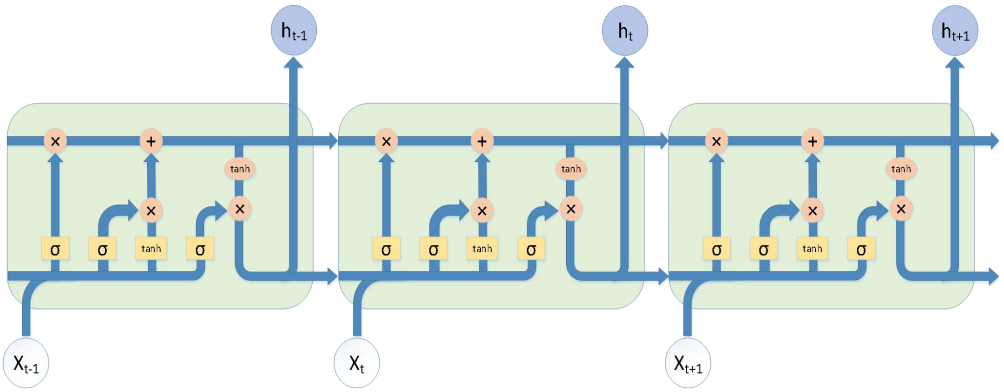


图 3.3 LSTM 神经网络模型结构图

LSTM 的前向传播算法与 RNN 类似，输入都是一个长度为 T 的时间序列数据:

$$f_t = \sigma(W_f \bullet [h_{t-1}, x_t] + b_f) \tag{3.14}$$

$$i_t = \sigma(W_i \bullet [h_{t-1}, x_t] + b_i) \tag{3.15}$$

$$\tilde{C}_t = \tanh(W_c \bullet [h_{t-1}, x_t] + b_c) \tag{3.16}$$

$$C_t = f_t * C_{t-1} + i_t * \tilde{C}_t \tag{3.17}$$

$$o_t = \sigma(W_o \bullet [h_{t-1}, x_t] + b_o) \tag{3.18}$$

$$h_t = o_t * \tanh(C_t) \tag{3.19}$$

公式(3.14)中f_t为遗忘门参数，1 表示“完全保留”，0 表示“完全舍弃”。公式(3.15)、(3.16)计算输入门的值。然后通过公式(3.17)丢弃需要丢弃的信息，加上在时间 t 输入的有用信息。公式(3.18)、(3.19)确定了将细胞状态的哪一个部分进行输出到下一层神经网络和下一个时刻。

LSTM 的时间后向传播算法与 RNN 模型中的时间后向传播算法中类似，从时间序列最后面的数值开始，逐步反向计算各参数的梯度，最后分别用各个时刻的梯度更新网络参数。首先计算记忆细胞输出对应的偏导数，再计算输出门偏导数，分别计算记忆细胞状态、遗忘门、输入门对应的偏导数，最终使用梯度下降法更新模型连接权重。

3.3.3 适用于自相关序列的长短周期模型

时序数列是一个时间上有序的数据集合，每个数据都对应相应的时间。时间序列模型会假设过去数据随着时间变化的规律同样适用于将来，可以通过学习历史数据的信息来预测将来的数据。

在常规的时序数列预测问题中，假设时间序列为$(y_1,y_2,y_3 \ldots y_t)$，其中 y_t 表示 t 时刻的时序数列的值。通过$(y_{m-k+1},y_{m-k+2} \ldots y_m)$来预测 y_{m+1} 的值，其中 k 表示每次预测使用的数据

步数。

根据时间颗粒度不同划分，可以定义以下四种预测类型：实时预测、短期预测、中期预测、长期预测[40]。实时预测要求数据之间时间间隔最短，可以用来建立在线实时预测系统。短期预测的时间间隔通常为一小时到数小时之间，常用来做最优化控制或异常检测。中期预测时间间隔为数天，可用来指导资源规划。长期预测的时间间隔为数月到数年，可以为制定策略、经济投资做参考。

自相关系数 r_k 用于表示时间序列自身与滞后 k 个时期数据相比的相关程度[41]：

$$r_k = \frac{\sum_{t=1}^{T-k}(y_t - \bar{y})(y_{t+k} - \bar{y})}{\sum_{t=1}^{T}(y_t - \bar{y})} \tag{3.20}$$

其中，$y_1,y_2...y_T$ 代表时间序列数据，T 表示数据集的总长度，$\bar{y}$ 表示序列的平均数。r_k 值越高，表示在周期 k 上，数据具有更好的自相关性，意味着数据集 y_i 在时间维度上以 k 为周期进行循环，不同周期内相同时刻的数据往往具有大致相同的数据特征。

传统的 RNN 时序数列预测模型通过学习数列的变化趋势，根据前 T 时间步的数据预测下一时刻。根据经验可知，网络流量数据在 24 小时和 7 天的周期上具有一定的自相关性，这意味着某一时刻前 24 小时或者 7 天的数据可以很好的表示当前时刻的数据特点。但是由于 24 小时或 7 天前的数据与前 T 时间步的数据并不能构成连续的时序关系，传统的 RNN 神经网络不适用于加入自相关特征进行建模。

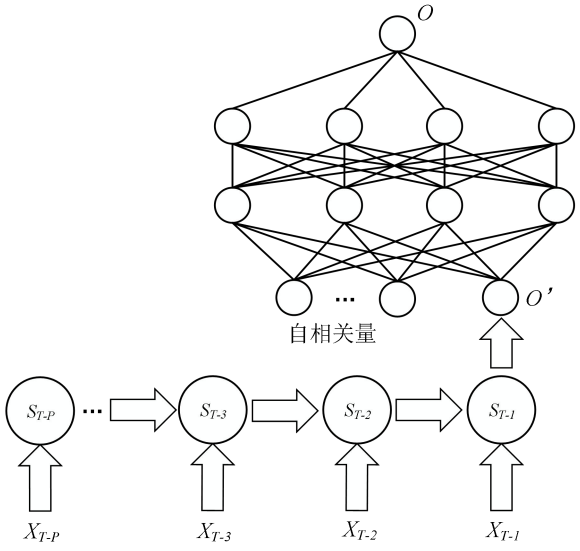


图 3.4 LSTM 与 ANN 结合的神经网络结构

本文将长短时记忆模型与人工神经网络（Artificial Neural Network, ANN）相结合，通过改进的 LSTM-ANN 神经网络，保持原有的 LSTM 网络不变，将预测出的值与一个自相关周期前的若干个值组成 ANN 网络的输入。通过训练 LSTM-ANN 网络，将网络

数据流量的自相关特征和时序特征结合起来，实现对网络数据流量精确预测。

3.4 实验与结果分析

3.4.1 数据集选取

为了验证网络流量预测模型的准确性，本文选用三个真实网络流量数据集进行实验。数据集 A 来自于欧洲某 11 个城市的网络服务提供商的网络流量历史数据。数据从 2004 年 6 月 7 日 06：57 采集到 2004 年 6 月 29 日 11：17，每 30 秒采集一次，数据如图 3.5。数据集 B 来自于英国教育和研究网络协会(United Kingdom education and research networking association, UKERNA)发布的用于学术研究的骨干网的网络流量历史数据。数据从 2004 年 11 月 19 日 9：30 采集到 2005 年 1 月 27 日 11：11，每 5 分钟采集一次，数据如图 3.6。数据集 C 来自于 2012 年中国教育网北京邮电大学骨干节点采集的网络流量数据，每一分钟采集一次，数据如图 3.7。

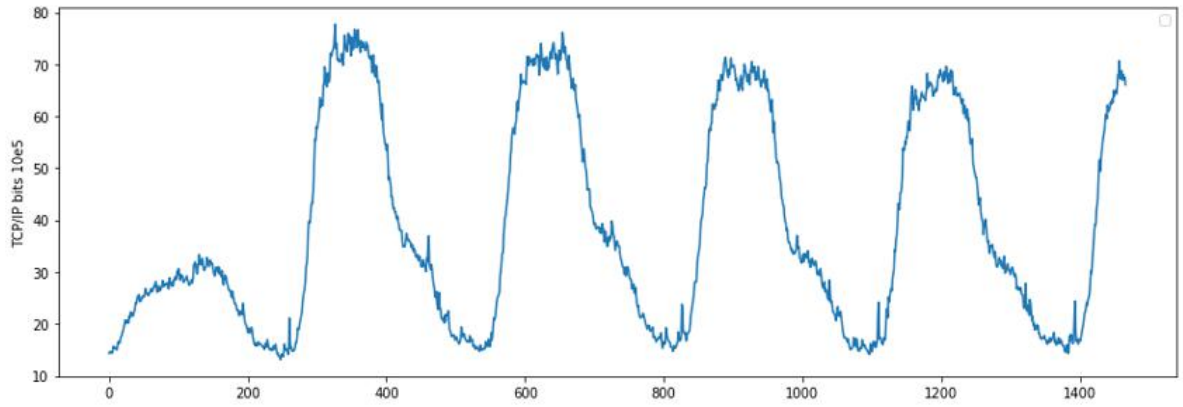


图 3.5 欧洲某 11 个城市的网络服务提供商网络流量数据

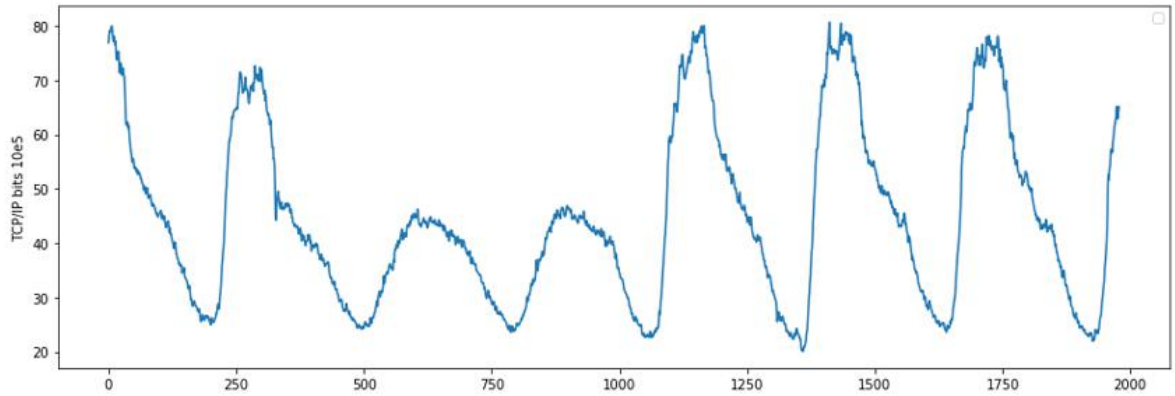


图 3.6 英国教育和研究网络协会网络流量数据

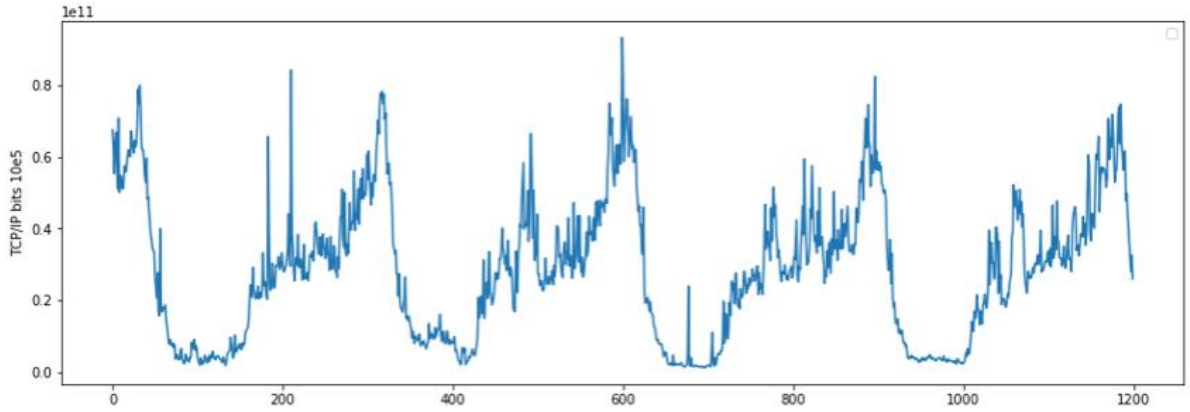


图 3.7 中国教育网北京邮电大学骨干节点网络流量数据

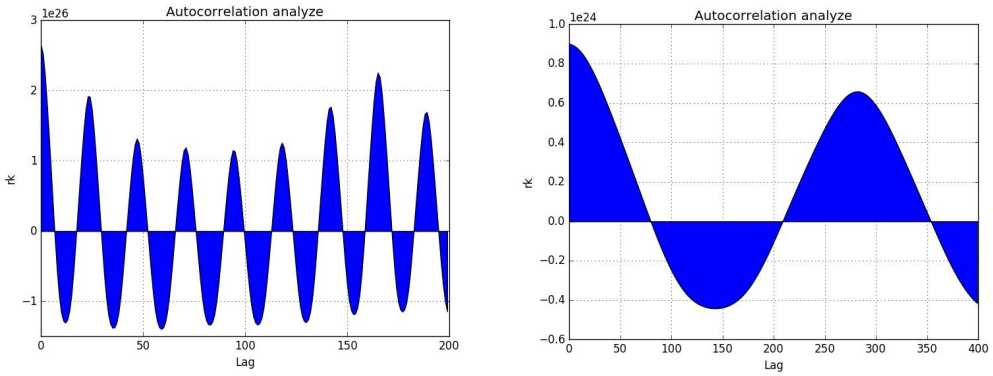
根据三份实验数据集采集的周期和时长的特点，本文只考虑前两种预测类型。参考 Cortez 对于时间颗粒度的选取方法[42]，分别采用 5 分钟、1 小时作为实时预测、短期预测的时间颗粒度。划分训练集和测试集时，将数据集的前 2/3 用于训练，后 1/3 用于测试模型的精确度。

为了评估预测模型的准确性，本文采用平均绝对百分比误差(Mean Absolute Percentage Error, MAPE)作为模型的评价指标[43]：

$$MAPE = \frac{1}{N}\sum_{n=1}^{N}\left|\frac{y_n - \hat{y}_n}{y_n}\right| \times 100\% \tag{3.21}$$

如公式(3.21)所示，平均绝对百分比误差是统计上常用于评估模型预测精确度的一个指标，其中 y 表示数据的真实值，$\hat{y}$ 表示预测值，N 为测试集大小。误差值越小，预测的结果越好。

3.4.2 自相关性分析



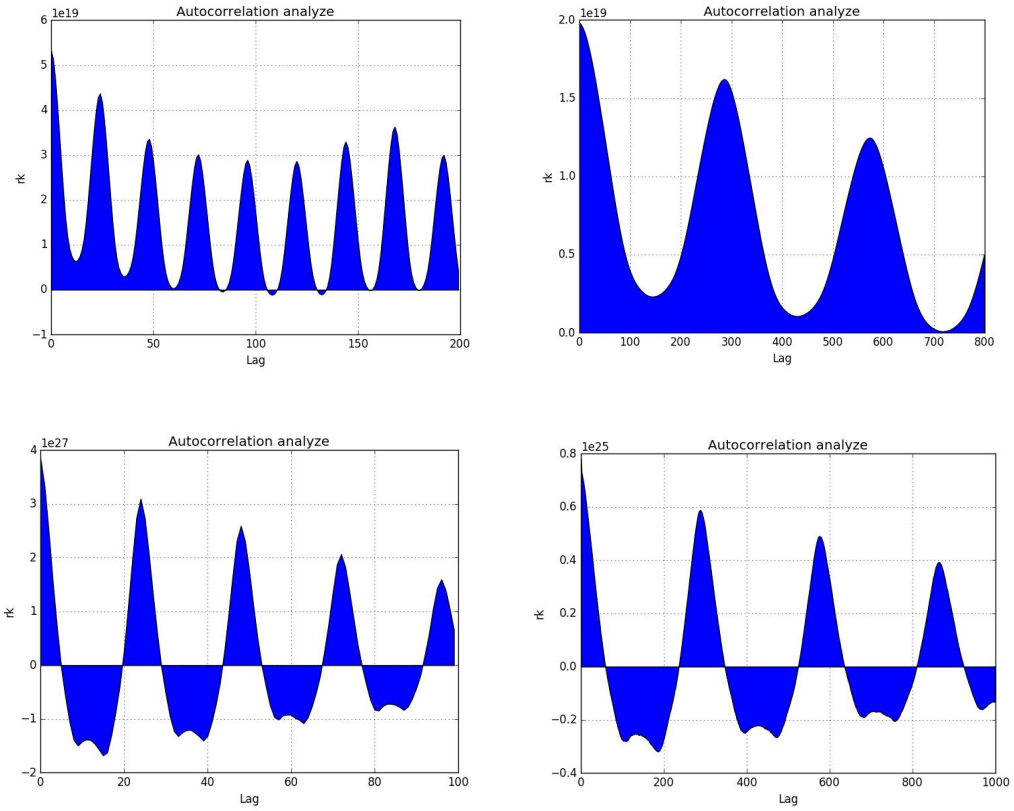


图 3.8 数据集自相关性分析

数据集分别为 A1h, A5min, B1h, B5min, C1h, C5min

从图 3.8 数据集自相关性分析中可以看到，对于以 5 分钟为时间颗粒度的数据中，k=288 时自相关程度较高，对于以 1 小时为颗粒度的数据中，k=24 和 k=168 时为两个峰值。说明网络流量的自相关周期为 24 小时和 7 天，符合大众对于网络流量数据的特性判断。

3.4.3 实验过程

本文实现了回声状态网络(Echo state network, ESN)作为 LSTM 模型的对比，ESN 也是 RNN 的一种变形，在 2001 年被研究者提出，在时序数列预测问题上表现出较为突出的优越性。由输入层，隐藏层，输出层组成，并且在隐藏层到隐藏层之间有一个连接，用来保留前面时刻的输入在网络中留下的信息。不同于 RNN 模型，ESN 的输入层到隐藏层、隐藏层到隐藏层的连接权值是随机初始化，并且固定不变。在训练的过程中，只需要去训练隐藏层到输出层的连接权值，所以 ESN 的训练速度非常快。根据经验以及实验比较，ESN 模型的参数最终选定为表 3.1 所示：

表 3.1 回声状态网络参数

In Size	1

Reservoir Size	1000
Leaking Rate	0.3
Spectral Radius	0.136
Training Algorithm	Ridge Regression

在 LSTM 模型中，为了防止过拟合，采用 Dropout 对模型中神经元进行处理。Dropout 最早由 Hintion 提出[44]，主要思想是让神经网络在训练过程中随机让一部分神经元不工作。经过实验证明，这样可以有效的防止过拟合，提高模型的泛化能力。Zaremba 改进了 Dropout[45]，使其可以应用于 RNN。在本文的模型中，训练过程中每个神经元不工作的概率为 10%。LSTM 神经网络参数如下：

表 3.2 LSTM 神经网络参数

Dropout Fraction	10%
Time Steps	10
RNN Units	200
RNN Layers	1
Dense Units	[10,10]
Batch Size	10

在 LSTM-ANN 模型中，LSTM 神经网络的参数与表格 3.2 一致。根据数据集时间颗粒度的不同，选用不同的自相关量与 LSTM 前馈算法计算出的值一起输入到 ANN 中。例如，在时间颗粒度为 5 分钟的数据集中，x_{i-287},x_{i-288},x_{i-289} 可以很好的反映 x_i 的数据量特征。所以，将 x_{i-287},x_{i-288},x_{i-289} 三个数据输入实时网络流量预测模型用来预测 x_i 的流量。在时间颗粒度为 1 小时的数据集中，将 x_{i-23},x_{i-24},x_{i-25} 三个数据输入实时网络流量预测模型用来预测 x_i 的流量。

3.4.4 实验结果分析

三种算法的平均绝对百分比误差如下表所示：

表 3.3 平均绝对百分比误差分析

数据	时间颗粒度	ESN	LSTM	LSTM-ANN
数据集 A	5min	2.81%	1.41%	1.39%
	1h	7.63%	4.69%	4.51%
数据集 B	5min	3.50%	3.29%	1.32%
	1h	5.04%	4.47%	2.80%

数据集 C	5min	4.41%	12.04%	——
	1h	0.59%	14.00%	——

由于数据集长度的限制，数据集 C 划分的测试集满足不了 LSTM-ANN 训练、预测的要求。从表中可以看出 LSTM 可以进行有效的网络流量预测，在两个数据集的不同时间颗粒度上均有良好的表现。根据自相关特性优化的 LSTM-ANN 模型，将数据的自相关量加入其中，在时间颗粒度为 5min 时，对精确度有微小的提升，但对于时间颗粒度为 1h 的情况下，预测精确度提升明显。这说明，自相关量弥补了 LSTM 需要大量数据训练的不足，在粗时间颗粒度或者数据量较少的情况下，LSTM-ANN 模型相对于传统 LSTM 有一定优势。

LSTM-ANN 的预测效果如图 3.9-3.12 所示：

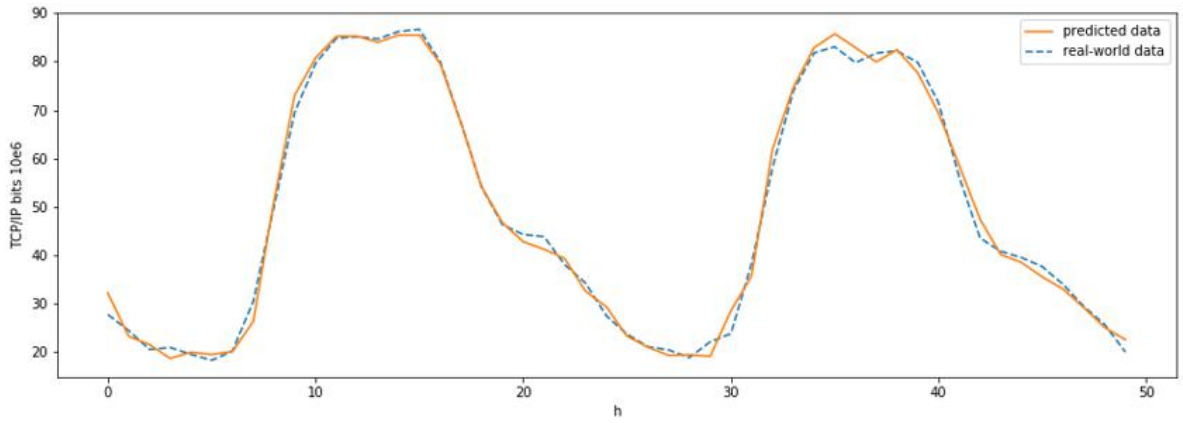


图 3.9 A1h 预测效果

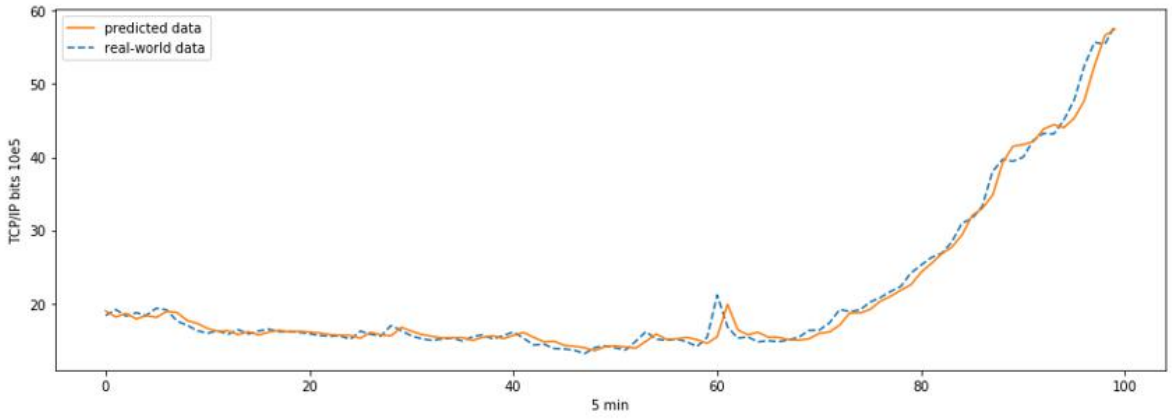


图 3.10 A5min 预测效果

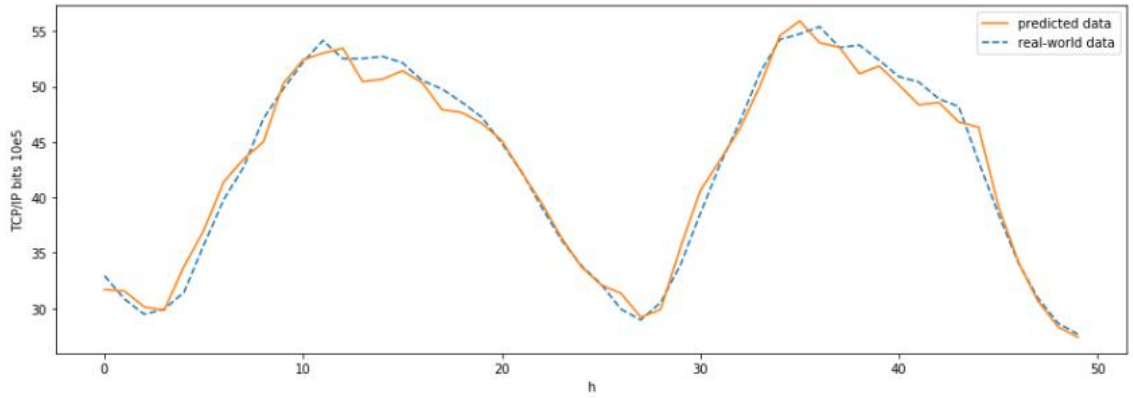


图 3.11 B1h 预测效果

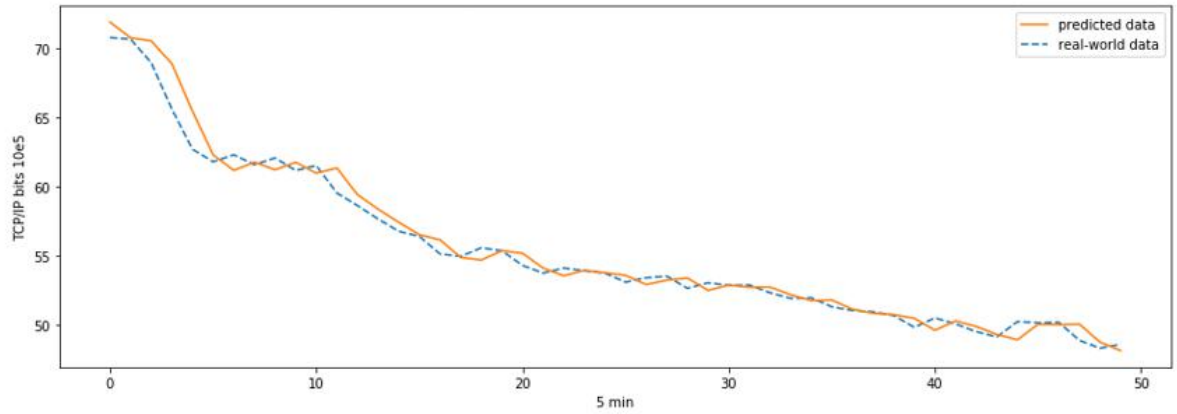


图 3.12 B5min 预测效果

3.5 本章小结

本章针对具有自相关性的网络流量进行研究，总结出一种利用 LSTM 与 ANN 结合的适用于自相关序列的长短周期神经网络模型。通过对国内外现实采集的多份数据集进行实验，实验结果表明 LSTM 可以很好的作为时序数列预测模型使用，提供优于其他传统模型的精确度。并且在考虑自相关特性之后，LSTM 与 ANN 结合的神经网络在粗时间粒度数据集的准确性方面具有一定的优势，但是对于数据量的大小有一定的要求。高精度的网络流量预测对于处理可能遇到的网络拥塞、异常攻击等情况提供了一定支持。由于 RNN 的变种繁复多样，LSTM 只是其中之一，同样热门并在很多问题中有良好表现的还有 GRU 等 RNN 结构。下一步的研究工作是考察不同的 RNN 结构，结合网络流量数据的特点，探索进一步提高预测精度的可能性。

4　基于深度学习的网络业务流量分类研究

4.1 引言

随着人们对互联网需求的增长，促使各种各样不同类型的互联网应用飞速发展，随之而来的是网络复杂程度的上升，这给网络安全及提高网络服务质量带来巨大的挑战。近年来，各种网络安全问题层出不穷，多种网络盗窃、网络监听、病毒攻击等问题时有发生，给网络安全行业工作人员敲响了警钟。为了能够更好的监督控制网络，为互联网用户提供更好的上网体验，网络资源提供商或者网络安全研究人员需要识别不同应用产生的流量，了解各类应用所占带宽比例，还需要识别出哪些是正常网络流量哪些是恶意网络流量，有效的网络流量分类技术就成为实现以上目标的重要技术基础[46]。

4.2 网络流量数据相关技术

计算机网络是一个庞大的集合，其体系结构非常复杂，需要有一个适当的方法来研究、设计和实现网络体系结构。网络体系结构是指对构成计算机网络的各组成部分及计算机网络本身所必须实现的功能进行定义，即网络体系结构是计算机网络中不同的层次、各层的协议以及层次间接口的集合。本节介绍网络的分层模型，和基于网络分层模型的网络数据格式。

4.2.1 网络分层模型

为了简化问题，降低网络和网络协议设计的复杂性，使网络便于维护，提高网络运行效率。目前网络设计一般采用层次结构，各层次结构相对独立，实现的功能相对独立。除最底层和最高层之外，中间每一层都是利用下一层提供的服务完成本层功能，同时为上一层提供一定的服务，并对上一层屏蔽本层服务实现的细节。分层的优点是层与层之间只在层次间接口处关联，层间耦合最小。网络体系结构具有可分层的特性，同样网络协议也具有可分层的特性，各层协议相互协调，构成一个整体，通常称为协议集或协议族。

目前流行的两大网络体系结构是开放互联参考模型 OSI/RM（Open System Interconnection Reference Model）以及 TCP/IP 参考模型两种。

OSI 模型将网络通信的工作划分为 7 层，这 7 层由高到低分别是应用层（Application Layer）、表示层（Presentation Layer）、会话层（Session Layer）、传输层（Transport Layer）、网络层（Network Layer）、数据链路层（Data Link Layer）和物理层（Physical Layer），OSI 模型如图 4.1 所示。从第 5 层到第 7 层属于 OSI 模型的底层部分，负责创建网络通信链接的链路，通常被称为通信子网；第 1 层到第 3 层是 OSI 模型的高层部分，具体负责端到端的会话服务、数据通信、加密/解密等功能，通常称之为资源子网；第 4 层是

OSI 参考模型的高层与底层之间的连接层，起着在底层和高层之间承上启下的作用，是 OSI 参考模型中从低到高第一个端到端的层次。每层完成自身的功能，为上层提供相应的服务，网络通信需要在网络模型中自下而上（在接收端）或者自上而下（在发送端）双向进行。但是，并不是网络中的所有通信都需要经过 OSI 的全部 7 层，例如，物理接口之间的连接、中继器与中继器之间的连接只需在物理层中进行；路由器与路由器之间的连接只需要在网络层以下的三层——通信子网中进行。

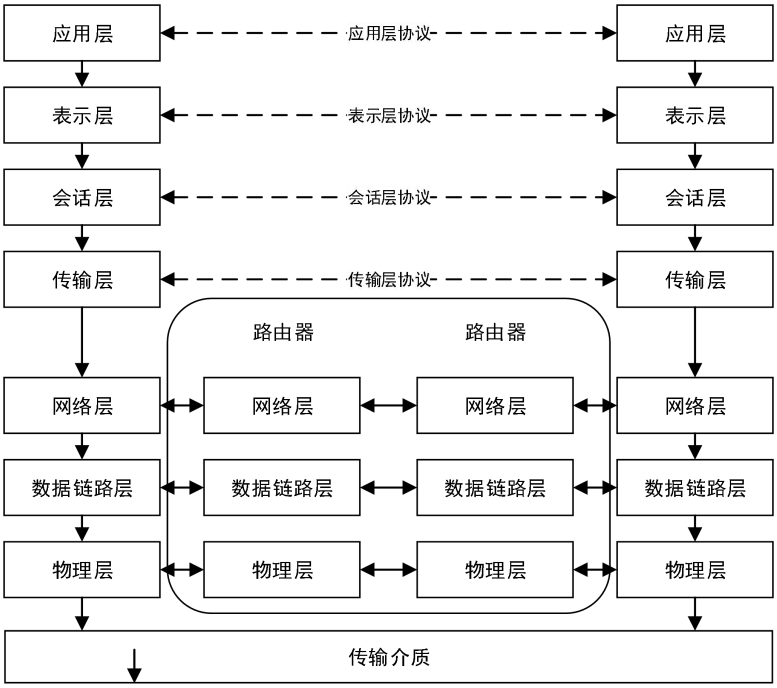


图 4.1 OSI 模型图

TCP/IP 体系结构分为四层，如图 4.2 所示，其体系结构模型自下而上分别是网络接口层、网络层、传输层和应用层。其中虚线框中的数据链路层和物理层严格说并不属于 TCP/IP 体系结构，但却被 TCP/IP 的网络接口层很好地调用。

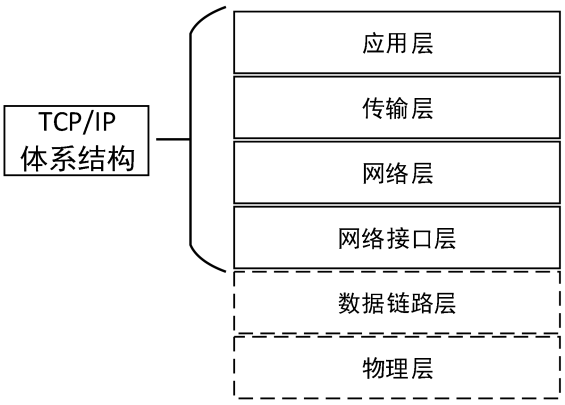


图 4.2 TCP 模型图

网络中发送方应用程序的数据总是从最上层开始，沿着网络分层结构层层逐步向下，每经过一层都在数据上加上该层的信息，对数据进行一些封装处理，例如打包或者编码，最终由物理层进行传输；相应的接收方的物理层接收到网络数据后，逐层向上，每层都会对数据进行一些处理，例如解码或者解包，来实现该层的服务，最终由应用层发到需要到达的应用程序中去。数据在网络层次模型上传输的示意图如图 4.3 所示。

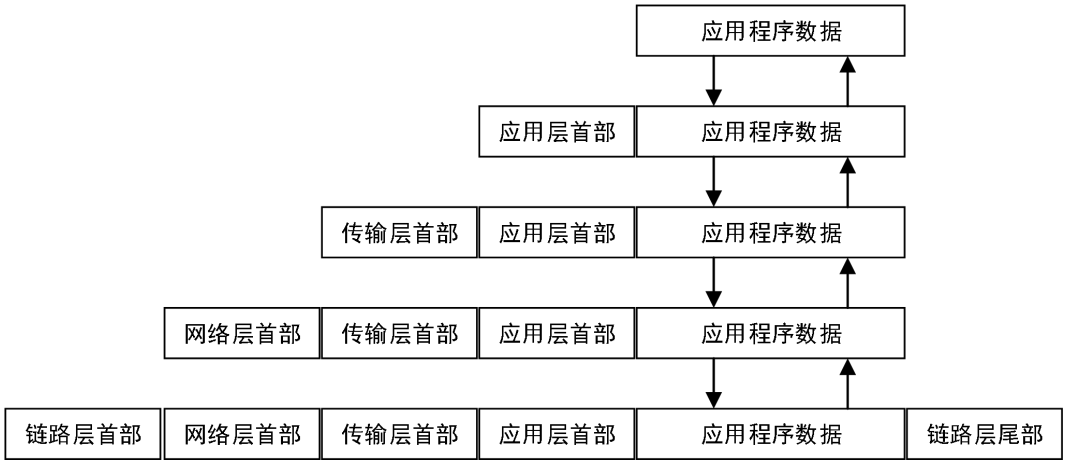


图 4.3 数据在网络分层模型中传输示意图

4.2.2 网络流量数据

在现有的网络流量业务分类问题研究中，绝大多数都采用网络特征数据集作为实验的分类对象，其中比较经典的有 Moore-set、KDD-CUP99、NSL-KDD 等。Moore-set 是剑桥大学的 Moore 教授在某个网络中心分若干个时间段采集的，并利用流量构造、数据挖掘等技术得到了多个数据集合，称为 entry01-entry10。这十个数据集中共有 377526 个网络流样本，这些样本覆盖了网络中常用的应用。在该数据集中，Moore 教授从流样本中提取 248 种流属性，并对每个流样本进行了应用类别的标记[33]。KDD-CUP1999 是 KDD 竞赛在 1999 年举行时采用的数据集，它是由林肯实验室在 1998 年通过建立模拟美国空军局域网的网络环境中，收集了 9 周时间的 TCPdump 网络连接数据，并在此基

础上进行特征分析得到一个具有 41 个特征的数据集。NSL-KDD 为 KDD-CUP99 数据集的优化[47]。有一些研究在这些数据集上取得了不错的分类准确度，其中比较有代表性的有 Gao et al.[48]使用的深度信念网络（Deep Belief Network, DBN），其在 KDD-CUP99 数据集上取得了优于支持向量机（SVM）和人工神经网络（ANN）的效果。但是这些基于人工提取或者传统特征提取算法的数据集并没有充分发挥深度学习模型可以自主学习数据特征的优势，因此本文采用了通过网络嗅探技术得到的原始流量数据集作为深度学习模型的输入。

以太网作为一种使用方便，原理简单的互联网技术已经成为网络传输的主流。以太网的数据帧是对网络分层结构中数据链路层的封装，网络层的数据包被加上链路层的帧头数据和帧尾就构成了可以被数据链路层识别的数据帧。网络流量嗅探工具获取的就是数据帧里面携带的数据包，在格式上又以 PCAP 格式作为主流。

PCAP 文件的内部结构如图 4.4 所示。每个 PCAP 文件存在一个 PCAP 文件头和多个数据包，每个数据包又都有自己的数据包头和数据包内容。首先，PCAP 文件的文件头占据 24 字节的大小，分别由文件识别头、主版本号、次版本号、当地标准时间、时间戳精度、最大储存长度、链路类型组成。每个数据包头占据 16 字节大小，分别由高位时间戳、低位时间戳、数据区长度和实际长度组成，每部分数据占据 4 个字节的长度。数据包内容长度为包头中定义的抓包长度，这个长度后面的内容就是下一个数据包的内容，网络中各层的协议定义了数据包的内容。

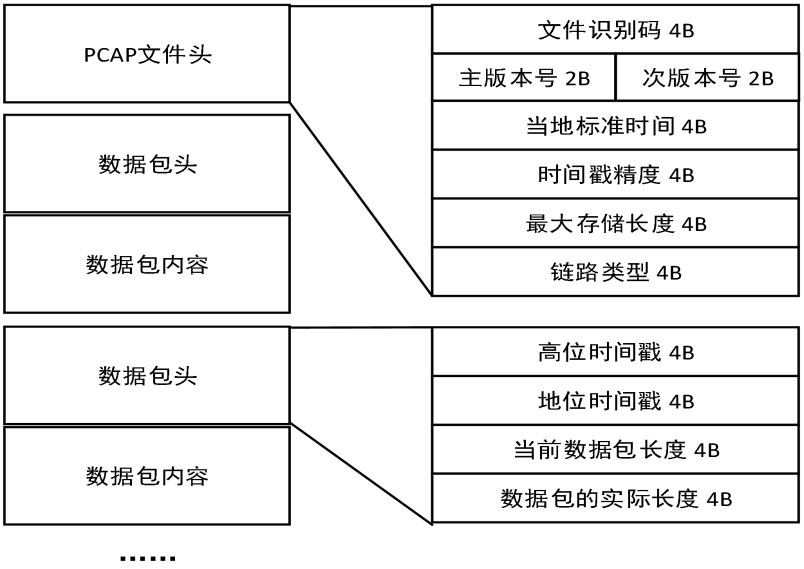


图 4.4 PCAP 文件结构示意图

在网络数据中最终分析的是数据包中的内容，下面以一个 HTTP 数据包为例来讲解

数据包的构成。数据包的最前面是数据链路层头部，也称以太帧头部，大小一共 14 字节。前 6 个字节为目的 MAC 地址，后 6 个字节为发送者 MAC 地址，最后的两个字节表示网络协议信息。在图 4.5 中，“0x0800”表示 IP 协议，所以以太帧头部之后是 IP 协议数据。IP 协议第一个字节“0x45”表示 IP 协议的版位为 IPv4，长度为 20 字节；后面一个字节表示服务类型；3、4 字节表示首部和数据的长度；5、6 字节是数据被分片后的标识，以方便正确的重组原来的数据报；7、8 字节分为前 3 位为标志位和后 13 位片偏移；9 字节表示数据报在网络中的寿命；10 字节表示数据报携带的协议类型；11、12 字节表示首位校验和，以方便判断数据报的保留与丢弃；13-16 字节为发送者的 IP 地址；17-20 字节为接受者的 IP 地址。由于 IP 数据报中携带的协议类型为“0x06”，所以后面为 TCP 协议。TCP 协议中，1、2 字节是源端口；3、4 字节是目标端口；5-8 字节是序列号；9-12 字节为确认号；13、14 字节的前 4 位是数据偏移字段，中间 6 位是保留字段，后面 6 位分别是紧急、确认、推送等标志位；15、16 字节是窗口大小，表示在未收到确认时，可以发送的最大字节数；17、18 是校验和；19、20 为紧急指针；21 字节往后是选项和填充位。TCP 协议后面就是 TCP 协议携带的 HTTP 协议内容，即网络中最终的应用数据。数据包数据的示意图如下图所示：

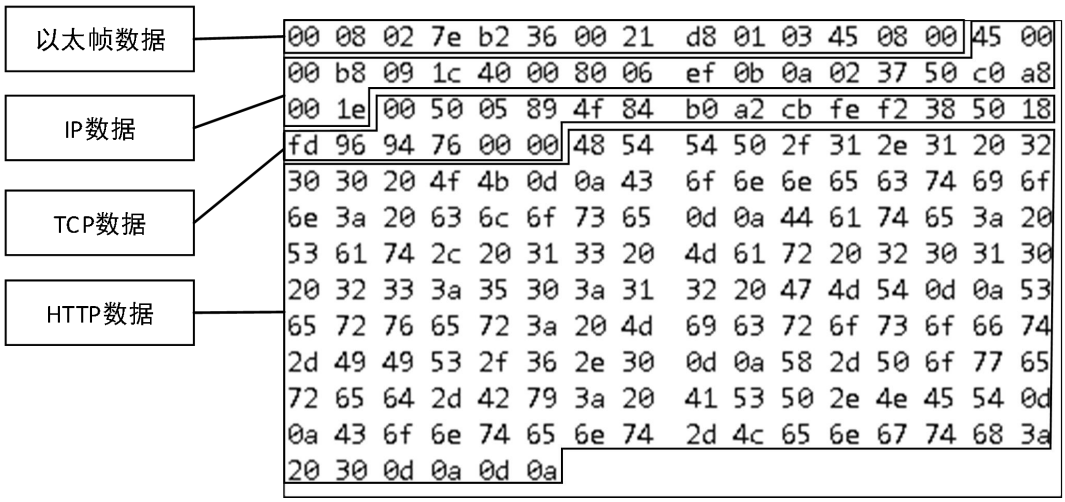


图 4.5 网络数据包示意图

4.3 卷积神经网络

卷积神经网络（Convolutional Neural Network, CNN）的早期模型称为神经认知机，是受到视觉系统神经机制的启发而演变出的一种生物物理模型。卷积神经网络可以看作是一种特殊的多层感知器或前馈神经网络，具有局部连接、权值共享的特点，其中大量神经元按照一定方式组织起来，以对视野中的交叠区域产生反应。自从卷积神经网络在深度学习领域登场之后，就很快取得了突飞猛进的发展，屡屡在图像分类和识别、目标定位与检测的大规模竞赛中名列前茅、战绩辉煌，甚至对提高机器在棋类博弈方面的智

能水平提高也发挥了不可估量的作用。

4.3.1 卷积神经网络的经典模型

研究者最初受到视觉神经机制的启发，针对二维图形识别问题设计的一种多层感知机，这就是卷积神经网络的原型。该模型在图像平移、缩放和倾斜的情况下都表现出良好的不变性。

1962 年，Hubel 等人通过对猫视觉系统皮层细胞的研究，提出了感受野（receptive field）的概念[49]。在 1984 年，研究者 Fukushima 在感受野的基础上，又提出了神经认知机（Neocognitron）模型[50]，它被学界认为是首次卷积神经网络的实现。1989 年，LeCun 等人首次使用了权值共享（weight sharing）的概念[51]。1998 年 LeCun 等人将卷积层与池化层结合，构成了卷积神经网络的主要结构，这是现代卷积神经网络的主要模型结构的来源（LeNet）[52]。

标准的卷积神经网络通常具有较深的神经网络结构，一般由输入层、卷积层（Convolutional Later），下采样层（Downsampling Layer）、全连接层（Fully-Connected Layer）以及输出层组成[51]。其中，卷积层也称为“检测层”（Detection Layer），下采样层又叫作为“池化层”（Pooling Layer）。图 4.6 为一种经典卷积神经网络的网络结构，其中池化层是互相不重叠的。输入层是一个矩阵，如一幅图像的灰度值矩阵。作为一种特殊的前馈网络，卷积层和下采样层就是特殊的隐含层，而输出层意外的其他层如全连接层是普通隐含层。卷积神经网络里的各层一般有不同的计算方式，其中的权值需要一个学习过程来调优。

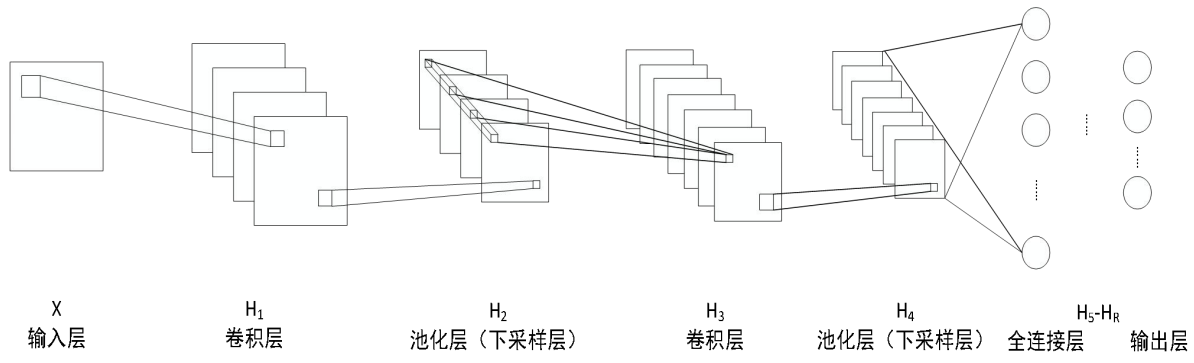


图 4.6 经典卷积神经网络结构图

图中卷积神经网络的第一个隐含层用 H_1 表示，由于 H_1 是通过卷积来计算的，所以又称为卷积层。如果分别用 $h_{1,\alpha}$ 和 $W^{1,\alpha}$ 表示 H_1 的第 α 个卷积面和第 α 个卷积核，那么 $h_{1,\alpha}$ 实际上是利用输入 x 与 $W^{1,\alpha}$ 进行卷积运算“ $\tilde{*}$ ”再加上偏置 $b^{1,\alpha}$ 得到的结果，即：

$$\mathrm{h}_{1,a} = f(u_{1,\alpha}) = f(C^{1,\alpha} + b^{1,\alpha}) = f(x \tilde{*} W^{1,\alpha} + b^{1,\alpha}) \tag{4.1}$$

卷积层H_1由所有卷积面$\mathrm{h}_{1,\alpha}$构成，即$\mathrm{H}_1=(h_{1,\alpha})$。卷积神经网络的第2个隐含层用$\mathrm{H}_2$表示，由于$\mathrm{H}_2$是通过对$\mathrm{H}_1$下采样来计算的，所以又成为下采样层或者池化层。使用$\mathrm{h}_{2,\alpha}$表示H_2的第α个下采样面，那么$h_{2,\alpha}$与$h_{1,\alpha}$的关系为：

$$\mathrm{h}_{2,\alpha}=g(\beta_2 \bullet down(h_{1,\alpha})+\gamma_2) \tag{4.2}$$

其中权值β_2一般取值为1，偏置γ_2一般取值为0矩阵。g()一般取为恒等线性函数$g(x)=x$。下采样层H_2由所有下采样面$\mathrm{h}_{2,\alpha}$构成，即$\mathrm{H}_2=(h_{2,\alpha})$。

卷积神经网络的第3个隐含层用H_3表示。由于H_3是通过从H_2选择多个下采样面和多个卷积核来计算的，所以也成为卷积层。从H_2中选择r个下采样面，分别用$\mathrm{h}_{2,\alpha_i}(1\le i\le r)$来表示，相应的卷积核用$\mathrm{W}_{\alpha_i}^{3,(\alpha_1,\alpha_2\ldots\alpha_r)}$表示，那么可以在卷积层$\mathrm{H}_3$构造一个卷积面如下公式（4.3），其中$\omega=(\alpha_1,\alpha_2\ldots\alpha_r)$。

$$h_{3,\omega}=h_{3,(\alpha_1,\alpha_2\ldots\alpha_r)}=f(\sum_{i=1}^{r}h_{2,\alpha_i}\ \tilde{*}\ W_{\alpha_i}^{3,\omega}+b^{3,\omega}) \tag{4.3}$$

卷机神经网络的第4个隐含层用H_4表示，是对H_3进行下采样计算得到的，所以也是一个下采样层。使用$\mathrm{h}_{4,\alpha}$表示H_4的第α个下采样面，那么$h_{4,\alpha}$与$h_{3,\alpha}$的关系为：

$$\mathrm{h}_{4,\alpha}=g(\beta_4 \bullet down(h_{3,\alpha})+\gamma_4) \tag{4.4}$$

其中权值β_4一般取值为1，偏置γ_4一般取值为0矩阵。下采样层H_4由所有下采样面$\mathrm{h}_{4,\alpha}$构成，即$\mathrm{H}_4=(h_{4,\alpha})$。

全连接的各层分别用$\mathrm{H}_5\sim\mathrm{H}_\mathrm{R}$来表示，主要用来分类这些层实际上构成一个普通的多层前馈网络，其中的激活函数一般采用*sigmoid*函数。最后一层H_R称为输出层，一般使用*softmax*进行输出。

4.3.2 卷积神经网络的学习算法

由于卷积神经网络在本质上是一种特殊的多层前馈网络，因此它的权值和偏置是可以用反向传播法进行学习和训练的。对于第l个样本，标准卷积神经网络从输入到输出的计算过程如公式（4.5）：

$$\begin{aligned}
&h_{1,\alpha}^{l}=f(u_{1,\alpha}^{l})=f(x^{l}\ \tilde{*}\ W^{1,\alpha}+b^{1,\alpha})\\
&h_{2,\alpha}^{l}=g(\beta_2 \bullet down(h_{1,\alpha}^{l})+\gamma_2)\\
&h_{3,(\alpha_1,\alpha_2\ldots\alpha_r)}^{l}=f(\sum_{i=1}^{r}h_{2,\alpha_i}^{l}\ \tilde{*}\ W_{\alpha_i}^{3,\omega}+b^{3,\omega})\\
&h_{4,(\alpha_1,\alpha_2\ldots\alpha_r)}^{l}=g(\beta_4 \bullet down(h_{3,(\alpha_1,\alpha_2\ldots\alpha_r)}^{l})+\gamma_4)\\
&H_4^{l}=(h_{4,(\alpha_1,\alpha_2\ldots\alpha_r)}^{l})\\
&H_k^{l}=\sigma(u_k^{l})=\sigma(W^{k}H_{k-1}^{l}+b^{k}),5\le k\le R
\end{aligned} \tag{4.5}$$

根据前向计算公式，可以推导出反向传播算法为：

算法：标准卷积网络的反向传播算法

输入：训练集 $S=\{(x^l,y^l),1\le l\le N\}$ 、网络结构、层数 R

输出：网络参数 $W^{1,\alpha}$ 、 $b^{1,\alpha}$ 、 $W_{\alpha_i}^{3,\omega}$ 、 $b^{3,\omega}$ 、 W^k 、 $b^k(5\le k\le R)$

1. 随机初始化所有权值和偏置
2. 计算 $H_0^l=x^l,u_{1,\alpha}^l=x^l\tilde{*}W^{1,\alpha}+b^{1,\alpha},h_{1,\alpha}^l=f(u_{1,\alpha}^l)$
3. 计算 $u_{2,\alpha}^l=\beta_2\bullet down(h_{1,\alpha}^l)+\gamma_2,h_{2,\alpha}^l=g(u_{2,\alpha}^l)$
4. 计算 $u_{3,\omega}^l=\sum_{i=1}^{r}h_{2,\alpha i}^l\tilde{*}W_{\alpha i}^{3,\omega}+b^{3,\omega},h_{3,\omega}^l=f(u_{3,\omega}^l)$
5. 计算 $u_{4,\omega}^l=\beta_4\bullet down(h_{3,\omega}^l)+\gamma_4,h_{4,\omega}^l=g(u_{4,\omega}^l)$
6. 令 $H_4^l=(h_{4,\omega}^l)$
7. 计算 $u_k^l=W^kH_{k-1}^l+b^k,H_k^l=\sigma(u_k^l),5\le k\le R$
8. 令 $o^l=H_R^l$,计算 $\delta_R^l=(o^l-y^l)\circ\sigma'(u_R^l)$
9. 计算 $\delta_k^l=[(W^{k+1})^T\delta_{k+1}^l]\circ\sigma'(u_k^l),5\le k\le R-1$
10. 计算 $\delta_4^l=[(W^5)^T\delta_5^l]\circ g'(u_4^l),\delta_4^l=(\delta_{4,\omega}^l)$
11. 计算 $\delta_{3,\omega}^l=\beta(f'(u_{3,\omega}^l\circ up(\delta_{4,\omega}^l)))$
12. 计算 $\delta_{2,\alpha i}^l=[\delta_{3,\omega}^l\tilde{*}rot180(W_{\alpha i}^{3,\omega})]\circ g'(u_{2,\alpha i}^l)$
13. 计算 $\delta_{1,\alpha}^l=\beta(f'(u_{1,\alpha}^l))\circ up(\delta_{2,\alpha}^l)$

14. 计算
$$\begin{cases} \dfrac{\partial L_N}{\partial W^k}=\sum_{l=1}^{N}\delta_k^l(H_{k-1}^l)^T, \dfrac{\partial L_N}{\partial b^k}=\sum_{l=1}^{N}\delta_k^l, 5\le k\le R \\ \dfrac{\partial L_N}{\partial W_{\alpha i}^{3,\omega}}=\sum_{l=1}^{N}h_{2,\alpha i}^l \tilde{*} \delta_{3,\omega}^l \\ \dfrac{\partial L_N}{\partial b^{3,\omega}}=\sum_{l=1}^{N}\delta_{3,\omega}^l \\ \dfrac{\partial L_N}{\partial W^{1,\alpha}}=\sum_{l=1}^{N}x^l \tilde{*} \delta_{1,\alpha}^l, \dfrac{\partial L_N}{\partial b^{1,\alpha}}=\sum_{l=1}^{N}\delta_{1,\alpha}^l \end{cases}$$

15. 更新所有的网络参数

其中 *rot180()*表示把一个矩阵水平翻转一次再垂直翻转一次。

4.4 实验与结果分析

4.4.1 实验数据集

如 Z Wang 等人[53]，很多关于流量业务分类的研究都选择自己抓取流量数据进行试验。由于在日常使用网络的过程中，恶意流量数据倾斜度非常的高，在互联网流量中占据极少一部分，导致对于恶意流量特征的学习非常依赖于训练的数据样本。使用私有数据集进行实验，会使实验内容无法被其他研究者重复，减弱算法在有效性上的说服力。为了避免这样的问题，本文恶意流量数据选用公开的 CTU-13 数据集。CTU-13 数据集由一组在实际网络环境中对 13 个不同的恶意软件捕获数据组成。捕获包括僵尸网络、正常流量和背景流量。僵尸网络流量来自受感染的主机，正常流量来自经过验证的正常主机，背景流量是所有其他流量的汇总。数据集以流为单位进行标记，是目前最大和最丰富的僵尸网络数据集之一。

本文使用的正常流量来自加拿大网络安全研究所网站提供的带有应用标签的 ISCX 数据集。其 VPN-nonVPN 数据集中，研究者定义了一组任务，确保数据集的多样性和数量上足够丰富。数据集中为用户 Alice 和 Bob 创建账户，以便使用 Skype，Facebook 等服务。在 VPN-nonVPN 数据集中，一共含有七种不同业务类型的流量数据，它们分别是浏览网页、电子邮件、使用 Skype，ICQ 等聊天工具聊天、Youtube 视频、FTP 文件传输、Skype 网络电话、Bittorrent 下载。所有类别的网络流量数据格式为 PCAP，该数据集的内容如表 4.1：

表 4.1 ISCX VPN-nonVPN 数据集内容

类别	内容
Web Browsing	FireFox and Chrome
Email	SMPTS, POP3S and IMAPS

Chat	ICQ, AIM, 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)

在数据的预处理环节中，首先要将原始的网络流量数据处理为合适的颗粒度大小，选择一次网络流产生的流量数据作为深度学习模型中的一个数据样本。首先定义一个五元组 X：源 IP、目标 IP、源端口、目标端口和传输协议。在本文使用的原始网络数据中，每个文件看成是若干数据帧的集合：$P=\{P_1,P_2...P_N\}$，每一个数据帧被定义为 $P_i=(x_i,b_i,t_i)$，其中 x 代表着一个五元组，b 代表着这个数据帧的大小，t 代表这个数据帧被传输的起始时间。一个网络流被定义为若干数据帧的集合 $f=\{P_1,P_2...P_N\}$，使用 $f=(x,b,d,t)$来表示，这个集合中的所有数据帧 P_i 具有这样的性质：他们拥有相同的五元组，这个数据流的大小 b 是数据帧大小 b 的集合，这个数据流持续的时间 d 是最后一个数据帧传输时间减去第一个数据帧的传输起始时间，数据流起始时间是第一个数据帧传输的起始时间。一个原始网络流量数据可以标识为 $F=\{f_1,f_2...f_N\}$。

在一次有效的网络流中，传输层携带的上层协议信息被称为其有效载荷（Payload），但由于网络不稳定、网络传送数据缺失等情况，常会有可能出现不携带 Payload 的无效网络流的情况。不携带应用层的网络数据对于网络业务分类来说是没有意义的，所以在数据预处理环节中应当根据是否存在有效载荷来筛选网络流，来达到使数据尽可能有效的目的。

由于网络流数据中的 IP 地址、MAC 地址对于网络流的业务分类没有关系，在传统统计学习模型中，为了防止他们对模型的影响，常常将数据中的所有 IP 地址和 MAC 地址删除或者随机化处理。然而深度学习模型通常具有自动提取有效特征，忽略无效特征的特性，所以可以将每份数据集处理为携带 IP、MAC 信息和不携带 IP、MAC 进行对比试验。在本文中，将这两类数据分别称之为 DATA1 与 DATA2。

数据集的预处理过程如下图 4.7 所示：

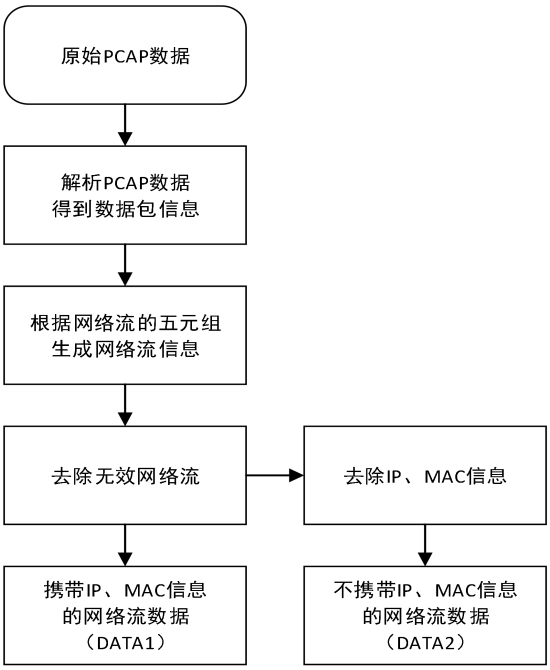


图 4.7 网络流量业务分类预处理流程图

经过处理后的数据集，一共 113084 条，其中按照正常/异常分为两大类，每大类又有 5 种具体的流量类型。所有数据的出处和流量类型的名称，在原数据集种的编号以及数据包数量网络流数量汇总情况如表 4.2：

表 4.2 网络流量数据

	名称	数据集编号	数据包数量	网络流数量
CTU-13 数据集的恶意流量	Neris	42	323154	12469
	Geodo	119-2	273478	53657
	Donbot	47	24764	4533
	Virut	54	440625	32818
	Murlo	49	85735	8523
ISCX 数据集的正常流量	Email	email1a/email1b /email2a/email2b	75079	243
	Skype	skype_chat1a /skype_chat1b	107865	164
	Spotify	spotify1/spotify2 /spotify3/spotify4	41254	127
	Bittorrent	vpn_bittorrent	422096	234
	Youtube	youtube1-6	120548	316

为了达到不同网络流量业务分类和恶意流量检测的两种目的，本文将学习任务分为两个场景：（1）仅作正常流量和恶意流量的分类，这是一个二分类问题。可以用于在网

络应用环境中进行恶意流量的检测和排查。（2）对所有网络业务进行分类，在本文中将测试集中的数据划分为 10 类，这是一个多分类问题。可以用于网络服务提供商或者网络管理员进行网络业务统计和管理。

两种学习场景的示意图如下图所示：

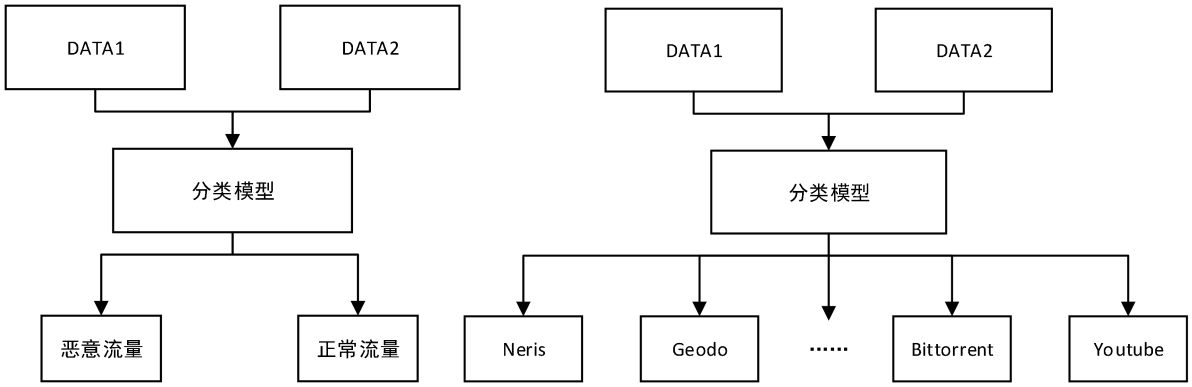


图 4.8 网络流量数据分类的两种场景

4.4.2 网络数据图像化

CNN 模型近几年在图像的分类、识别、边缘检测等方面都有成熟的应用。例如为了识别手写字符，将 CNN 模型在 MNIST 手写数字图像数据集上进行训练和测试，每一个手写数字图像都是由 28*28 的 256 级灰度像素点组成。在网络数据中，每一个字节都可以表示成为十进制[0, 255]区间中的数字，这刚好对应 256 灰度，所以本文可以将网络数据进行图像化。将网络数据图像化的过程如下图所示：

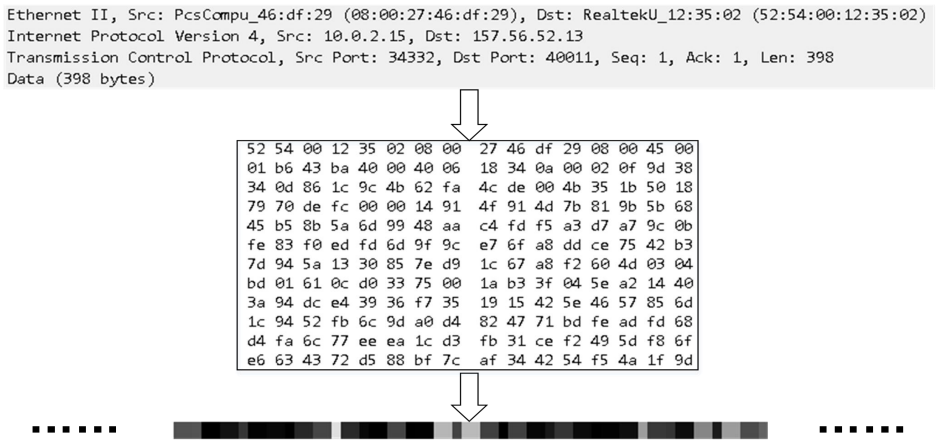


图 4.9 网络流量数据图形化过程

为了直观的感受不同类别数据之间的差异，本文将预处理好的数据进行图像化。在网络流量数据中，一个字节表示为黑白图像中一个像素点的 256 级灰度，从而每一个数据流前 1024 字节的数据可以表示为一个 32*32 像素的图像。将本文选取的十类流量图

像化以及选取两类同类型的不同网络流进行图像化如图 4.10、4.11 所示：

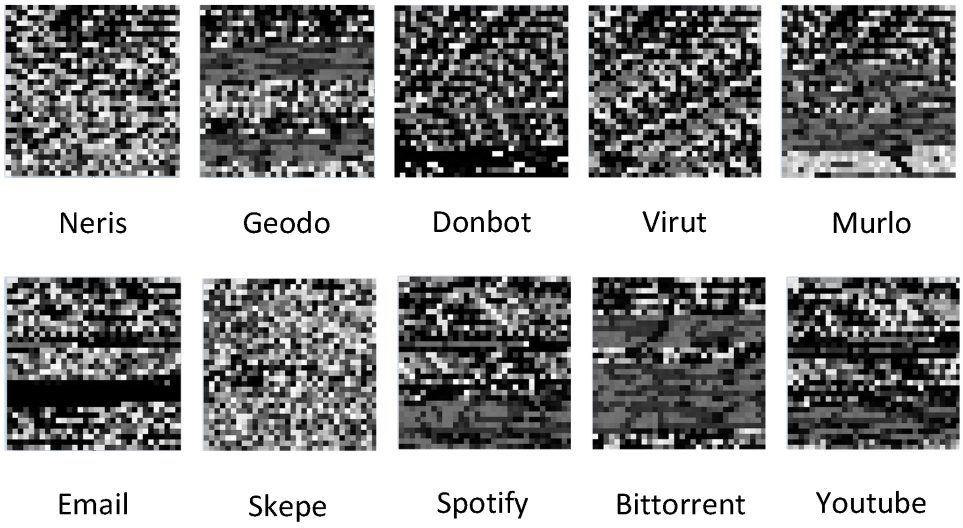


图 4.10 不同类型的网络流量数据图形化结果

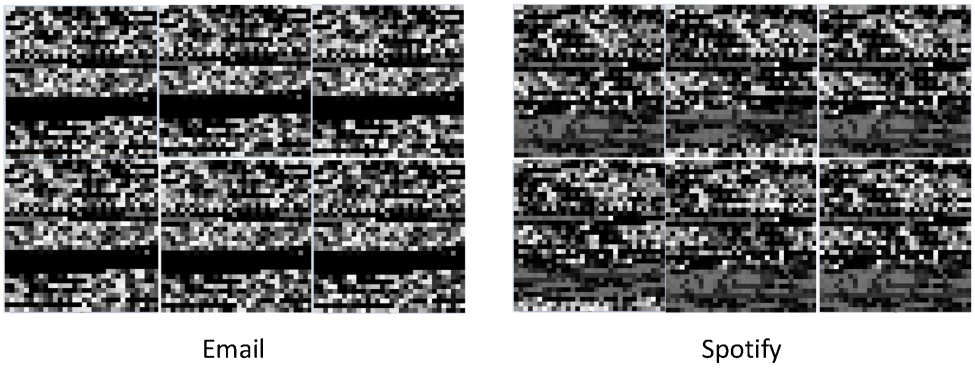


图 4.11 同种类型的网络流量数据图形化结果

从图中可以明显的看到，不同类型的流量具有明显不同的图形化特征，直观的表明了在网络流量分析的研究中是可以将 CNN 对于图像特征提取的优越性与网络流量数据分类问题结合起来的。

4.4.3 构建 CNN 模型

在使用卷积神经网络来完成图像分类问题中，Yann LeCun 为识别手写数字所设计的 LeNet-5 模型是卷积神经网络中最具代表性的模型之一。由于 LeNet-5 模型需要相同大小的输入，有研究结果表明[53]，网络流数据越靠前的数据对其分类结果影响越大。所以本文只截取每一网络流数据中的前 1024 字节，对于长度不足 1024 字节的数据，本文采取末尾补 0 的方式。LeNet-5 的网络结构如下图所示：

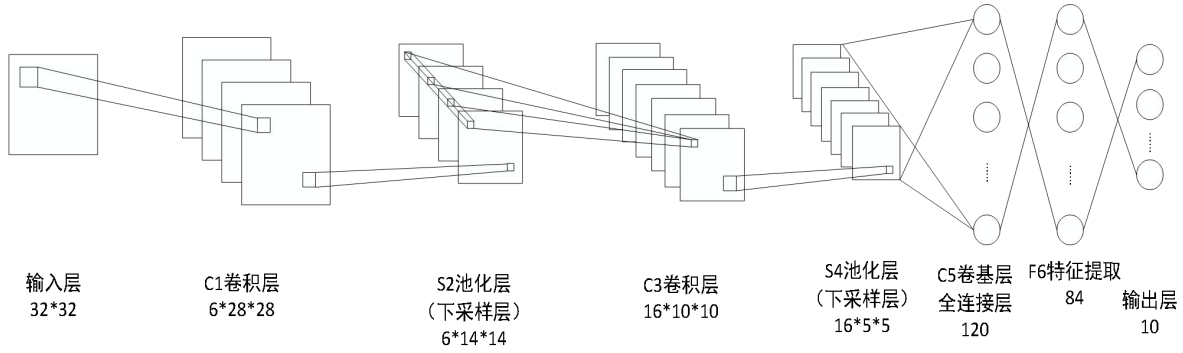


图 4.12 LeNet-5 网络结构图

在 LeNet-5 网络中，输入层是 32*32 大小的图像。*C1* 层是一个卷积层，卷积核大小为 5*5，卷积核每次移动距离为 1 个像素。*C1* 卷积层使用 6 个特征图谱，每一个特征图谱内的卷积核权值共享。*C1* 层每个特征图谱的大小为(32-5+1)*(32-5+1)即 28*28。*S2* 层是一个池化层，和 *C1* 一样，有 6 个特征图谱，每个特征图谱中的每个神经元都对应这 *C1* 层的 2*2 的区域。*S2* 层中每个神经元的值是 *C1* 层对应的 4 个神经元相加之和再乘以一个训练参数后加上这个特征图谱上的偏置参数，最后通过 *sigmoid* 激励函数计算得到。*C3* 也是一个卷积层，同样使用 5*5 的卷积核，卷积核每步移动 1 个像素。*C3* 拥有 16 个特征图谱，每个特征图谱的大小为(14-5+1)*(14-5+1)即 10*10。*C3* 层的每个特征图谱都是 *S2* 层对应的几个特征图谱计算得到的，他们的对应方式如表 4.3 所示：

表 4.3 LeCNN 中 C3 层与 S2 层特征图谱的对应关系

	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
0	√				√	√	√			√	√	√	√		√	√
1	√	√				√	√	√			√	√	√	√		√
2	√	√	√				√	√	√			√		√	√	√
3		√	√	√			√	√	√	√			√		√	√
4			√	√	√			√	√	√	√		√	√		√
5				√	√	√			√	√	√	√		√	√	√

S4 是一个池化层，有 16 个特征图谱，每个特征图谱中的每个神经元都对应这 *C3* 层的 2*2 的区域。*C5* 是一个卷积层，使用 5*5 的卷积核，*C5* 层特征图像的大小为 1*1，*C5* 层一共有 120 个特征图谱，每个神经元与 S4 层的全部 16 个特征图谱进行全链接。F6 是特征提取层，一共有 84 个神经元，每个神经元与 *C5* 层进行全连接。最后一层是输出层，在场景一中，定义十类输出，分别表示十类业务流量，在场景二中定义 2 类输出，分别为恶意流量与正常流量。对于这个针对二维图形化流量数据进行建模的卷积神经网络模型，在本文中称为 LeCNN。

虽然 LeNet-5 网络可以对图像的特征提取和分类有很好的效果，但是由于网络流量

数据的长度是可变的，所以需要一种能够支持可变长度的 CNN 分类模型。Ye Zhang 等人[54]为句子分类设计了一种 CNN 模型，并被其他的研究者应用在众多领域[55][56]，取得了不错的效果。该网络结构如下图所示：

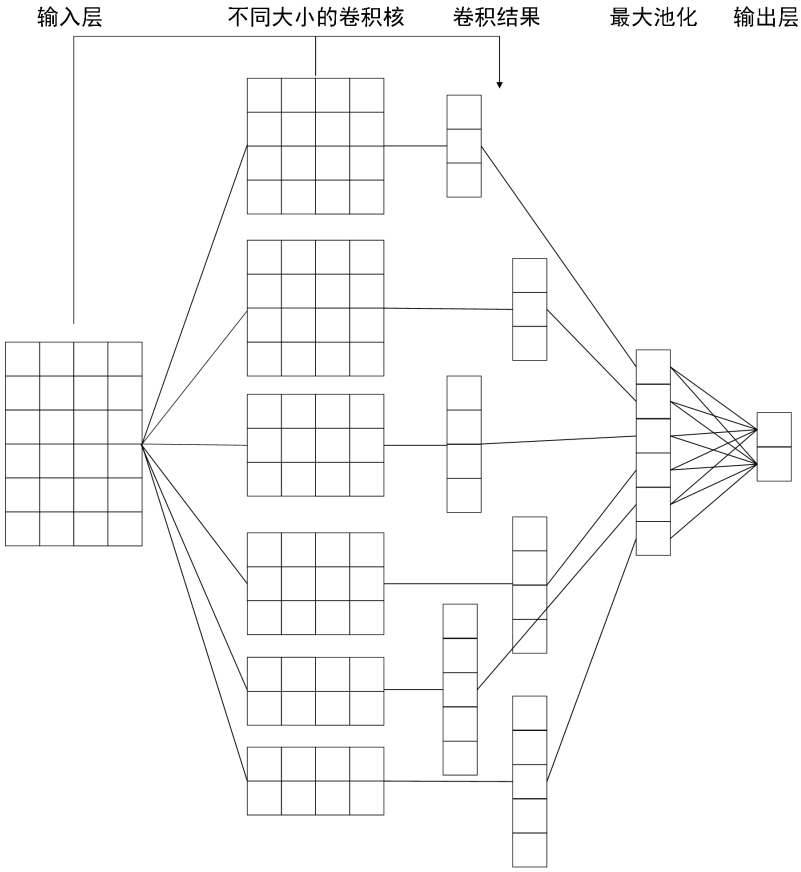


图 4.13 YeCNN 神经网络结构示意图

这个 CNN 模型的输入为一个 *n*k* 的矩阵，*n* 为一个网络流的字节数长度，矩阵的每一行为该字节数据对应的 One-Hot 表示，所以 *k* 的值为 256。*C1* 层为一个卷积层，每种卷积核的大小为 d*256，*d* 取值为{3,4,5}的数组。每种卷积核有 128 个，所以 *C1* 层对应的特征图谱的数量为 3*128。S2 为一个最大池化层，将每一个特征图谱中最大的值作为对应的神经元取值，所以不论输入的 *n* 为多少，*C1* 层的大小都是 384*1。输出层大小在场景一种为 10，在场景二种为 2，输出层与 *C1* 层采用全连接。这种可以适应不同长度流量数据输入的卷积神经网络在本文中称作 YeCNN。

模型的训练和分类采用 Google 公司的 Tensorflow 框架，运行在 Ubuntu 14.04 64 位电脑上，CPU 为 Intel E5，采用 NVIDIA GTX 1080 显卡加速运算。随机选取十分之一作为测试数据，其他为训练数据。训练中损失函数使用交叉熵，采用梯度下降法优化模型。学习效率设定为 0.01，经过 20 轮的学习得到最终的测试结果。

4.4.4 实验结果分析

为了评估模型的有效性，本文针对每一类分类使用准确率（Accuracy, A），精确率（Precision, P），召回率（Recall, R），F_1 值来作为模型分类有效性的评估标准，他们的定义如下：

$$\begin{aligned} A &= \frac{TP+TN}{TP+FP+FN+TN} \\ P &= \frac{TP}{TP+FP} \\ R &= \frac{TP}{TP+FN} \\ F_1 &= \frac{2PR}{P+R} \end{aligned} \tag{4.6}$$

其中 TP 为被判定为正样本中正确的样本数量，TN 为判定为负样本中正确的样本数量，FN 为判定为负样本中错误的样本数量，FP 为判定为正样本中错误的样本数量。

在场景一中，卷积神经网络将测试数据集分为正常流量（Positive）和恶意流量（Negative）两类，分别简写为 P 和 N。在含有 IP、MAC 信息的两份数据集上，使用 LeCNN 以及 YeCNN 两个不同的卷积神经网络模型在异常检测二分类场景下做分类任务。其中 DATA1 为去除 IP、MAC 信息的数据，DATA2 为包含 IP、MAC 信息的数据。图 4.14 表示这四个实验在这个场景下的准确率。

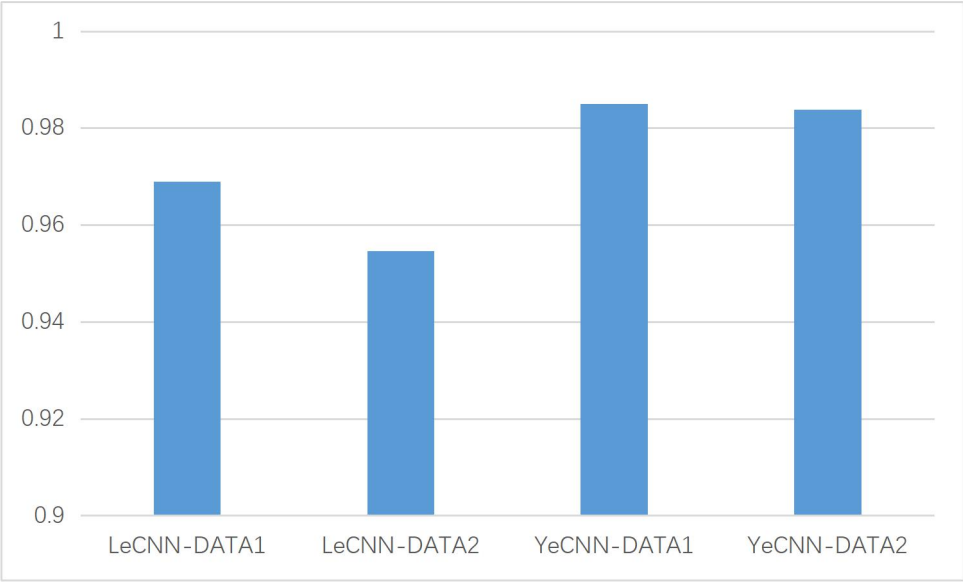


图 4.14 在场景一下的分类准确率

从准确率图表中可以看到，使用卷积神经网络对网络数据进行恶意流量检测是基本可行的，四个实验在这个场景下都达到了 95%以上的准确率。通过对比本文使用的两类

神经网络，YeCNN 在两个数据集中的准确率都要优于 LeCNN 模型，并在 DATA1 中取得了接近 98%的准确率。实验结果验证了 YeCNN 可以有效消除由于在网络流数据不足 1024 字节时，在末尾补齐“0”对分类结果产生的影响。包含了 IP、MAC 信息的数据在 YeCNN 模型中也对准确率影响较少，达到几乎持平的效果。可见使用 YeCNN 模型对网络流量数据进行恶意流量检测时，与训练任务无关的数据特征可以较少的影响分类结果。从而在实际应用中，可以选择不对这类信息做消除或是随机化处理，提高了数据预处理的效率。

在恶意流量分类场景中，恶意流量数据的精确率（P），召回率（R），F_1 值如图 4.15 所示：

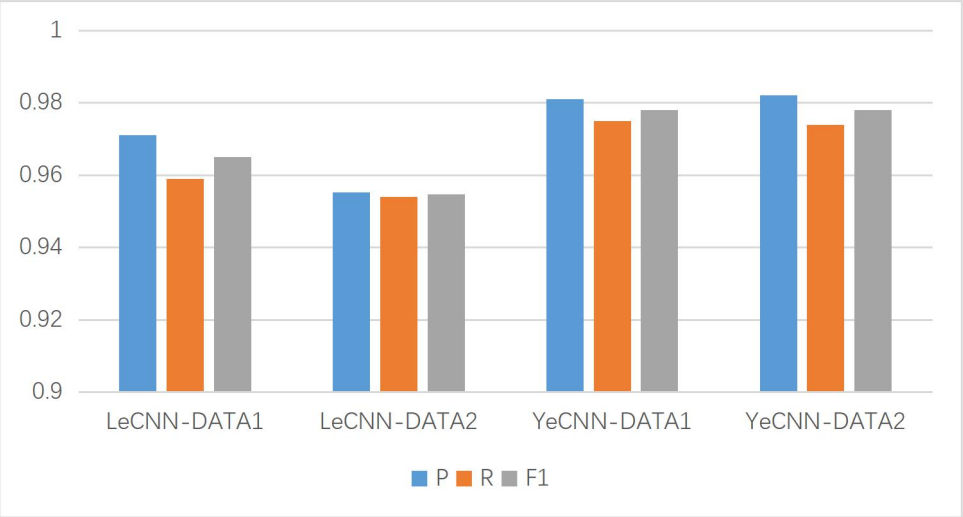


图 4.15 场景一下的精确率，召回率与 F_1 值

在网络流量多分类问题中，4 组实验的准确率如图 4.16：

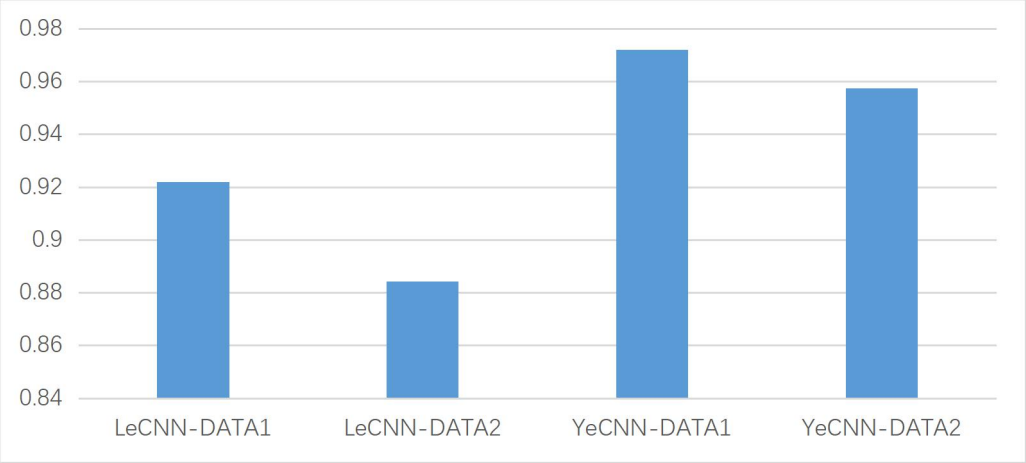


图 4.16 场景二中的分类准确率

在网络流量多分类场景中，使用 YeCNN 模型在 DATA2 上的各个类别的精确率（P），召回率（R），F_1 值如表 4.4 所示。本文还实现了栈式自编码网络（SAE）作为对比，网络中参数参考 M Lotfollahi 等人在文章中的网络结构[57]。

表 4.4 网络流量分类结果

类别	YeCNN			SAE		
	P	R	F_1	P	R	F_1
Neris	0.9732	0.9599	0.9665	0.9694	0.8901	0.9281
Geodo	0.9812	0.9817	0.9815	0.9810	0.9445	0.9624
Donbot	0.9395	0.9249	0.9321	0.9443	0.9735	0.9587
Virut	0.9934	0.9180	0.9542	0.9933	0.9101	0.9499
Murlo	0.9139	0.9718	0.9420	0.9195	0.9309	0.9292
Email	1	1	1	1	0.9583	0.9787
Skepe	1	1	1	1	1	1
Spotify	0.9231	0.9231	0.9231	0.8462	0.8462	0.8462
Bittorrent	0.9167	0.9565	0.9362	0.9091	0.8696	0.8889
Youtube	0.9063	0.9063	0.9063	0.9118	0.9688	0.9394
平均	0.9547	0.9542	0.9542	0.9475	0.9300	0.9381

从实验结果分析来看，基于深度学习的网络业务流量分类模型是有效的，并且在恶意流量检测方面具有实用作用。相对于手工特征提取的算法，本文算法具有深度学习模型特有的可以自动进行特征选取的性质。由于过去的研究大多基于自主抓取的数据，而且近几年提出的网络恶意流量检测算法准确率都能达到一个相当高的程度，所以本文着重比较本文提出的基于深度学习的网络流量业务分类算法与较其他算法在应用方面的一些优势。对比算法中选取了比较有代表性的 Han 和 Kamber 等人基于支持向量机的流量分类系统[58]，和目前大规模应用于企业中的基于报文签名的网络流分类方法。

表 4.5 本文算法与其他网络流量分类算法的比较

	本文算法	基于报文签名	SVM
算法效率	中等	很高	低
特征人工选取	不需要	需要	需要
在加密数据上的效果	有效	无效	无效
新兴恶意病毒对于算法的针对性	极不容易被针对	容易被针对	不容易被针对

在算法效率方面，虽然传统方法如 SVM 在特定的数据集上面具有较高的效率，但是数据集需要网络专家做特征提取预处理。基于深度学习特别是卷积神经网络的网络业务流量分类算法是直接针对网络链路层的二进制数据进行处理，无须做手工的特征提取，

所以效率上优于支持向量机算法。在特征的选取方面，由于如 Moore、KDD-CUP99 等数据集根据数据集作者对于网络数据处理方面不同的观点，选取了不同的数据特征，甚至在数量上大相径庭，所以很难保证这些数据特征包含的网络数据中足够多的有效信息。而本文算法是充分利用了深度学习模型的特性，在简化数据处理步骤的同时保证了特征的有效性。在日常应用中，很多网络数据由于敏感或者私有等目的，需要进行加密传输，如 Email 和 BitTorrent 等。针对 Email 数据的实验结果可以看出，本文模型在针对加密数据的分类问题上同样有效。而依赖于特征提取的数据集往往会由于数据加密的存在，不能正确的提取到真实有效的特征信息，从而在加密数据上效果不佳。在基于报文签名的检测算法甚至一些简单的机器学习算法应用过程中，恶意软件的开发者可以很容易针对入侵检测模型，对其软件的网络传输功能做出适当的调整以骗过检测算法。但是对于深度学习模型来说，由于算法的复杂性和不可解释性，恶意软件开发者很难针对深度学习模型进行调整。由于本文算法是使用原始二进制数据进行处理，避免了可能存在的伪造某一特征数据的情况，所以在算法的实用性上大大提高。

4.5 本章小结

本章研究了网络流量分析中的网络业务流量分类问题，采用深度学习的思想对其进行建模。

通过对过去研究工作的介绍，了解到传统网络流量业务识别技术无法应对目前复杂多变的网络环境。而近几年基于机器学习技术的网络流量业务分类方法仍旧依赖于网络专家对网络特征的提取，在实时性和实用性方面都存在不足。本章从网络流量数据中最原始的数据出发，只需简单的预处理就可以直接输入深度学习模型进行训练和分类。

将带有类别标签的网络流量进行可视化之后，可以直观的看到，不同类别的网络流量数据具有不同的图形特征，这为本文使用图形分类问题中常用的卷积神经网络提供了信心。然而，图像的分类问题需要固定大小的输入，而网络流数据的长度是长短不一的，所以本文又使用支持可变输入长度的卷积神经网络。通过实验验证，这两种类型的卷积神经网络在分类问题上都取得了不错的效果。

5 结论与展望

5.1 结论

本文针对前人在网络流量分析的研究上局限于只研究流量预测或者流量业务分类的某一方面，而提出从“量”和“质”的两个角度对网络流量进行研究。该思路结合了网络流量分析中最关键的两个问题，并在网络流量分析的应用中对这两个问题又进行了结合，他们的应用目的都可以划分为正常流量的管理和异常流量的检测两个部分。在网络流量分析的模型方面将深度学习引入到网络流量分析中，总结出了基于深度学习的网络流量的分析概念模型。针对于网络流量预测和网络流量分类这两个问题，本文分别考察了深度学习中两个重要的模型：递归神经网络和卷积神经网络以及他们的变种或者结构变形。

本文在网络流量分析模型方面的创新点如下：

（1）将长短时记忆模型引入网络流量预测问题中。选用递归神经网络中效果较好的长短时记忆模型，得到比传统统计学习方法更好的效果。

（2）在网络流量的预测中，考虑到网络流量自相关的特点，对长短时记忆模型进行了优化，在粗粒度数据中取得了更好的效果。

（3）将网络流数据图形化，直观的印证使用在图形分类问题上表现突出的卷积神经网络可以有效的对网络流量数据进行分类。并使用 LeNet-5 结构的网络，对网络流量数据进行业务分类。

（4）针对网络流量数据长度可变的特点，调整卷积神经网络的结构，使之可以适应不同长度的网络流量数据。并在实用性方面具有一定的优势。

5.2 展望

本文主要研究了基于深度学习的网络流量分析问题中模型的优化问题，所改进的模型相比较于前人的研究有一定的优势，但在模型的选择、优化方面仍有一定的进步空间，这也是本文下一步的工作：

（1）探寻递归神经网络在网络流量预测上的其他变形模型。递归神经网络变种多样，本文采用了较为流行的长短周期递归神经网络，还有如 GRU 等模型在网络流量预测上的表现还有待验证。

（2）进一步研究流量预测与数据流分类之间的联系。本文研究了数据包流量的预测和数据流的分类问题，由于网络流量数据自身具有强烈的时序特点，可以考虑建立统一的模型来实现网络流量的实时监控与异常检测等。例如在应用层面，如需建立网络异

常检测系统，需要从流量预测和流量分类两个方面分别建模，下一步工作会寻找一种统一的分析方法对应用流量进行分析，使用一个模型处理网络数据，进一步增强对网络流量的控制、预测、监控等效果。

参考文献

[1] 王刚. 网络流量分析技术在信息网管理中的应用[J]. 信息技术, 2015(4):161-164.

[2] 安航, 李启东, 王超超. 高校校园网络流量分析及流控策略[J]. 网络安全技术与应用, 2017(6):108-109.

[3] 熊刚, 孟姣, 曹自刚,等. 网络流量分类研究进展与展望[J]. 集成技术, 2012, 1(1):32-42.

[4] Fraleigh C, Moon S, Lyles B, et al. Packet-level traffic measurements from the Sprint IP backbone[J]. IEEE Network, 2003, 17(6):6-16.

[5] Barakat C, Thiran P, Iannaccone G, et al. Modeling Internet backbone traffic at the flow level[J]. IEEE Transactions on Signal Processing, 2003, 51(8):2111-2124.

[6] Tao H, Hui Z, Li Z, et al. A Methodology for Analyzing Backbone Network Traffic at Stream-Level[C]// International Conference on Communication Technology Volume 1 of. 2003:98-102 vol.1.

[7] Crovella M E, Bestavros A. Self-similarity in World Wide Web traffic: evidence and possible causes[C]// ACM SIGMETRICS International Conference on Measurement and Modeling of Computer Systems. ACM, 1996:160-169.

[8] Doulamis A D, Doulamis N D, Kollias S D. Nonlinear traffic modeling of VBR MPEG-2 video sources[C]// IEEE International Conference on Multimedia and Expo. IEEE, 2002:1318-1321 vol.3.

[9] Lopez-Guerrero M, Gallardo J R, Makrakis D, et al. Sensitivity of a network traffic prediction algorithm[C]// Communications, Computers and signal Processing, 2001. PACRIM. 2001 IEEE Pacific Rim Conference on. IEEE, 2001:615-618 vol.2.

[10]Sang A, Li S Q. A predictability analysis of network traffic[J]. Computer Networks, 2002, 39(4):329-345.

[11]Heady R, Luger G F, Maccabe A, et al. The architecture of a network level intrusion detection system[M]. University of New Mexico. Department of Computer Science. College of Engineering, 1990.

[12]Bro P V. A system for detecting network intruders in real-time[C]//Proc. 7th USENIX Security Symposium. 1998.

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

[14]Groschwitz N K, Polyzos G C. A time series model of long-term NSFNET backbone

traffic[C]// Communications, 1994. ICC '94, SUPERCOMM/ICC '94, Conference Record, 'Serving Humanity Through Communications.' IEEE International Conference on. IEEE, 1994:1400-1404 vol.3.

[15]Basu S, Mukherjee A, Klivansky S. Time series models for internet traffic[C]// Fifteenth Joint Conference of the IEEE Computer and Communications Societies Conference on the Conference on Computer Communications. IEEE Computer Society, 1996:611-620.

[16]舒炎泰, 王雷, 张连芳,等. 基于 FARIMA 模型的 Internet 网络业务预报[J]. 计算机学报, 2001, 24(1):46-54.

[17]Liu X, Fang X, Qin Z, et al. A Short-term forecasting algorithm for network traffic based on chaos theory and SVM[J]. Journal of network and systems management, 2011, 19(4): 427-447.

[18]Wang H, Hu D. Comparison of SVM and LS-SVM for regression[C]//Neural Networks and Brain, 2005. ICNN&B'05. International Conference on. IEEE, 2005, 1: 279-283.

[19]Junsong W, Jiukun W, Maohua Z, et al. Prediction of internet traffic based on Elman neural network[C]//Control and Decision Conference, 2009. CCDC'09. Chinese. IEEE, 2009: 1248-1252.

[20]Chen Y, Yang B, Meng Q. Small-time scale network traffic prediction based on flexible neural tree[J]. Applied Soft Computing, 2012, 12(1): 274-279.

[21]Cotton M, Eggert L, Touch J, et al. Internet assigned numbers authority (IANA) procedures for the management of the service name and transport protocol port number registry[R]. 2011.

[22]Park B C, Won Y J, Kim M S, et al. Towards automated application signature generation for traffic identification[C]//Network Operations and Management Symposium, 2008. NOMS 2008. IEEE. IEEE, 2008: 160‐167.

[23]Kang H J, Kim M S, Hong J W K. A method on multimedia service traffic monitoring and analysis[M]//Self‐Managing Distributed Systems. Springer Berlin Heidelberg, 2003: 93‐105.

[24]Van Der Merwe J, Caceres R, Chu Y, et al. Mmdump: A tool for monitoring Internet multimedia traffic[J]. ACM SIGCOMM Computer Communication Review, 2000, 30(5): 48‐59.

[25]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.

[26]Zuev D, Moore A W. Traffic classification using a statistical approach[M]//Passive and

Active Network Measurement. Springer Berlin Heidelberg, 2005: 321‐324.
[27]Zander S, Nguyen T, Armitage G. Self‐learning IP traffic classification based on statistical flow characteristics[M]//Passive and Active Network Measurement. Springer Berlin Heidelberg, 2005: 325‐328.
[28]Sen S, Wang J. Analyzing peer‐to‐peer traffic across large networks[J]. IEEE/ACM Transactions on Networking (ToN), 2004, 12(2): 219‐232.
[29]Tran Q A, Duan H, Li X. One-class support vector machine for anomaly network traffic detection[J]. China Education and Research Network (CERNET), Tsinghua University, Main Building, 2004, 310.
[30]Tan K M C, Collie B S. Detection and classification of TCP/IP network services[C]//Computer Security Applications Conference, 1997. Proceedings., 13th Annual. IEEE, 1997: 99‐107.
[31]Early J P, Brodley C E, Rosenberg C. Behavioral authentication of server flows[C]//Computer Security Applications Conference, 2003. Proceedings. 19th Annual. IEEE, 2003: 46‐55.
[32]Williams N, Zander S, Armitage G. A preliminary performance comparison of five machine learning algorithms for practical IP traffic flow classification[J]. ACM SIGCOMM Computer Communication Review, 2006, 36(5): 5‐16.
[33]Moore A, Zuev D, Crogan M. Discriminators for use in flow-based classification[J]. 2005.
[34]BABIARZ, R. and J. BEDO (2006) Internet traffic midterm forecasting: a pragmatic approach using statistical analysis tools, Lecture Notes on ComputerScience, 3976, 111–121
[35]ALARCON-AQUINO, V. and J. BARRIA (2006) Multiresolution FIR neural-network-based learning algorithm applied to network traffic prediction, IEEE Transactions on Systems, Man and Cybernetics – Part C, 36, 208–220.
[36]KRISHNAMURTHY, B., S. SEN, Y. ZHANG and Y. CHEN (2003) Sketch-based change detection: methods, evaluation, and applications, In Proceedings of Internet Measurment Conference (IMC'03), Miami, USA
[37]Bengio Y, Simard P, Frasconi P. Learning long-term dependencies with gradient descent is difficult[J]. IEEE Transactions on Neural Networks, 2002, 5(2):157-166.
[38]Hochreiter S. The vanishing gradient problem during learning recurrent neural nets and problem solutions[M]. World Scientific Publishing Co. Inc. 1998.
[39]Hochreiter S, Schmidhuber J. Long short-term memory[J]. Neural Computation, 1997,

9(8):1735.

[40]Ding X, Canu S, Denoeux T. Neural Network Based Models For Forecasting[C]// Neural Networks & Their Applications. 1995:243--252.

[41]Eric R. Ziegel. Time Series Analysis, Forecasting, and Control[M]. Holden-day, 1976.

[42]Cortez P, Rio M, Rocha M, et al. Multi-scale Internet traffic forecasting using neural networks and time series methods[J]. Expert Systems, 2012, 29(2):143–155.

[43]Jr A E F. Forecasting: Methods and Applications, by Spyros Makridakis, Steven C. Wheelwright and Rob J. Hyndman. Third edition. John Wiley and Sons, 1998, 642pp, ISBN 0-471-53233-9. £29.95, $90.65[J]. International Journal of Forecasting, 2002, 18(1):158-159.

[44]Hinton G E, Srivastava N, Krizhevsky A, et al. Improving neural networks by preventing co-adaptation of feature detectors[J]. Computer Science, 2012, 3(4):págs. 212-223.

[45]Zaremba W, Sutskever I, Vinyals O. Recurrent Neural Network Regularization[J]. Eprint Arxiv, 2014.

[46]林闯, 李寅, 万剑雄,等. 计算机网络服务质量优化方法研究综述[J]. 计算机学报, 2011, 34(1):1-14.

[47]M. Tavallaee, E. Bagheri, W. Lu and A. Ghorbani, "A detailed analysis of the KDD CUP 99 data set", Proc. 2009 IEEE Int. Conf. Comput. Intell. Security Defense Appl., pp. 53-58.

[48]N Gao, L Gao and Q Gao, "An Intrusion Detection Model Based on Deep Belief Networks", Advanced Cloud and Big Data (CBD) 2014 Second International Conference on, pp. 247-252.

[49]Hubel D H, Wiesel T N. Receptive fields, binocular interaction and functional architecture in the cat's visual cortex[J]. Journal of Physiology, 1962, 160(1):106.

[50]Fukushima K. Neocognitron: A self-organizing neural network model for a mechanism of pattern recognition unaffected by shift in position[J]. Biological Cybernetics, 1980, 36(4):193-202.

[51]Lecun Y, Boser B, Denker J S, et al. Backpropagation Applied to Handwritten Zip Code Recognition[J]. Neural Computation, 1989, 1(4):541-551.

[52]Lecun Y, Bottou L, Bengio Y, et al. Gradient-based learning applied to document recognition[J]. Proceedings of the IEEE, 1998, 86(11):2278-2324.

[53]Z. Wang, "The Applications of Deep Learning on Traffic Identification."https://goo.gl/WouIM6.

[54]Zhang Y, Wallace B. A Sensitivity Analysis of (and Practitioners' Guide to) Convolutional Neural Networks for Sentence Classification[J]. Computer Science, 2015.

[55]Schmaltz A, Kim Y, Rush A M, et al. Sentence-Level Grammatical Error Identification as

Sequence-to-Sequence Correction[J]. 2016.
[56] Wen Y, Zhang W, Luo R, et al. Learning text representation using recurrent convolutional neural network with highway layers[J]. 2016.
[57] 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.
[58] Han J, Kamber M. Data Mining: Concepts and Techniques, Morgan Kaufmann[J]. Machine Press, 2001 (in Chinese, 2006, 5(4):394-395.