

# 深度学习在SDN中的应用研究

李鹏飞, 邵维专

(四川大学 计算机学院, 四川 成都 610065)

**摘要:** 软件定义网络作为一种最新网络架构, 可通过软件编程的方式定义和控制网络, 其控制平面和转发平面分离及开放性可编程的特点, 为新型互联网体系结构研究提供了新的实验途径, 也极大地推动了下一代互联网的发展。深度学习相对于传统的机器学习有很多优点, 深度学习能够发现多层特征, 并能够将高层特征表示成更抽象的数据特征。深度学习网络模型因为具有多个隐藏层而具有很强的特征学习能力, 相对于机器学习模型来说具有很大的进步。随着深度学习的快速发展, 有必要在软件定义网络中引入深度学习, 推进软件定义网络的进一步发展。从架构、数据源、快速特征提取、深度学习算法选择和分析深度学习在SDN中的现有应用五个方面来说明深度学习在SDN中的应用。

**关键词:** 深度学习; 软件定义网络; 应用; 研究

中图分类号: TP393

文献标识码: A

文章编号: 1673-629X(2019)01-0001-05

doi: 10.3969/j.issn.1673-629X.2019.01.001

## Research on Application of Deep Learning in SDN

LI Peng-fei, SHAO Wei-zhuan

(School of Computer, Sichuan University, Chengdu 610065, China)

**Abstract:** Software defined networking (SDN), as an innovational network framework, can enable programmers to control and define the networks by software programming. Its features of separation of control plane and forwarding plane and open programmable have provided a new experimental approach for the study of new Internet architecture and greatly promoted the development of the next generation of Internet. Deep learning has many advantages over traditional machine learning, and it can discover multi-layer features and represent higher-level features as more abstract data features. The deep learning network model has a strong feature learning ability due to its multiple hidden layers, which is a great improvement compared with the machine learning model. With the rapid development of deep learning, it is necessary to introduce deep learning into the SDN which is promoted in further development. The application of deep learning in SDN is illustrated from five aspects including architecture, data source, rapid feature extraction, deep learning algorithm selection and analysis of existing applications of deep learning in SDN.

**Key words:** deep learning; software defined networking; application; research

## 0 引言

软件定义网络 (software defined networking, SDN)<sup>[1-2]</sup> 是近年来迅速发展起来的一种新型网络架构, 基本特点是将传统网络交换设备进行数据转发与控制的抽象分离。SDN网络架构与传统“漏斗”式网络架构相比, 最大的区别在于, SDN网络可以通过软件编写的方法实现对网络设备转发能力的灵活定义, 简而言之, SDN是一种可实现编程控制的网络。SDN架构一经推出就受到业界的高度认可, 并被认为是一种未来网络的终极解决方案。

2006年, Geoffrey Hinton等<sup>[3]</sup>提出了深度学习的

概念。深度学习是机器学习的一个分支, 是机器学习中表征学习方法的一类。在深度学习提出以前, 很多机器学习方法得到了广泛应用, 如支持向量机 (support vector machine, SVM)<sup>[4]</sup>、K最近邻法 (k-nearest neighbor, KNN)<sup>[5]</sup>、逻辑回归 (logistic regression)<sup>[6]</sup>、Boosting<sup>[7]</sup>等。然而, 这些模型的基本结构可以被理解为只有一层隐藏层或者没有隐藏层的简单人工神经网络。因此, 这些机器学习模型在处理多维数据时一般会存在一定的局限性, 通常这种局限性可称之为“维度爆炸”, 即高维的数据特征很难被完美地映射成低维的抽象数据特征。

收稿日期: 2018-01-13

修回日期: 2018-05-23

网络出版时间: 2018-09-21

基金项目: 国家自然科学基金面上项目 (61373091)

作者简介: 李鹏飞 (1990-), 男, 硕士研究生, 研究方向为软件定义网络、智慧协同网络。

网络出版地址: <http://www.cnki.net/kcms/detail/61.1450.TP.20180920.1535.008.html>

深度学习为解决机器学习的局限性创造了一个新的开端。深度学习算法是一个对特征学习的过程,在学习过程中能够发现多层特征,并能够将高层特征表示成更抽象的数据特征。深度学习网络模型因为具有多个隐藏层而具有很强的特征学习能力,相对于机器学习模型具有很大的进步。深度学习网络类似于大脑的神经网络,大脑中的神经网络由数千亿神经元连接而成,而深度学习网络同样由大量的人工神经元组成。深度学习网络中数据之间的关系由复杂的非线性模型来表示,然后通过训练大量的数据来确定数据之间的关系。目前,深度学习已经在语音识别、计算机视觉、自然语言处理等领域得到了广泛应用,由 Andrew Ng 领导的 Google Brain 团队已经构建了经过训练后可以识别高级概念图片的神经网络模型<sup>[8]</sup>。

SDN 作为一种新兴的网络体系结构,其智能化的发展需求逐渐被提出,而深度学习的发展正好契合了此需求。然而,目前深度学习并没有在 SDN 架构下的网络环境中得到广泛应用,因此文中旨在研究如何在 SDN 中引入深度学习,从而推动 SDN 的智能化发展。

### 1 架构

KDN<sup>[9]</sup>架构是各种机器学习方法应用于 SDN 的一种通用架构,所以对于深度学习应用于 SDN 同样适用。KDN 架构就是在 SDN 架构中引入知识平面<sup>[10]</sup>和管理平面,其架构如图 1 所示。

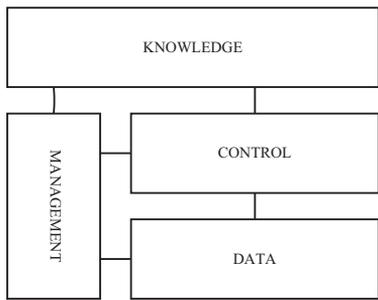


图 1 KDN 架构

数据平面:数据平面由转发设备组成,负责存储、转发和处理数据包。数据平面根据控制平面下发的流规则对数据报文进行解析、查表、过滤匹配以及端口映射等处理。

控制平面:控制平面由 SDN 控制器组成,逻辑集中的 SDN 控制器间交换操作状态进而更新流规则并将流规则以 Imperative 语言的形式通过南向接口下发到数据平面的转发设备。

管理平面:垂直于控制平面和数据平面,定义网络拓扑,收集并处理网络设备提供的信息,监控分析网络。

知识平面:使用深度学习处理方法处理管理平面收集

的信息,然后利用学习到的知识自动地或人为干预地制定具体的网络控制决定并将决定以 Intent 语言的形式下发到控制器,控制器再将 Intent 以 Imperative 语言的形式下发到转发设备。

KDN 工作流程如图 2 所示。

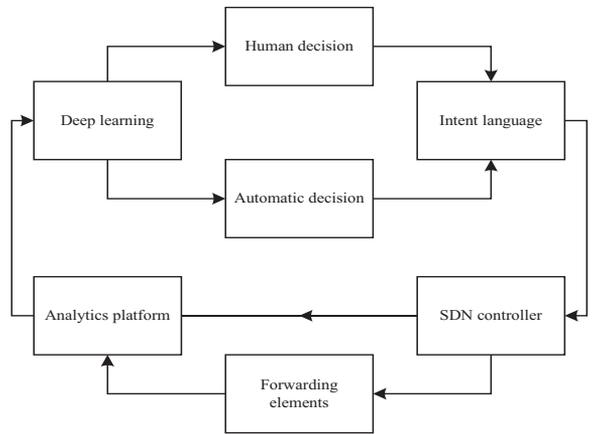


图 2 KDN 工作流程

分析平台实时监控数据平面获取细粒度的流量信息,并查询 SDN 控制器获取控制和管理状态,然后将收集到的数据提交给知识平面,知识平面使用深度学习算法学习网络行为并制定具体的网络控制决定并将决定以 Intent 语言的形式下发到控制器,控制器将接收到的 Intent 以 Imperative 语言的形式下发到转发设备。

### 2 数据集来源

#### 2.1 KDD99 和 NSL-KDD

KDD99 将攻击类型分为 4 类,然后又细分为 39 小类,每一类代表一种攻击类型,类型被标记在训练数据集每一行记录的最后一项。

KDD99 训练集中共出现了 22 个攻击类型,而剩下的 17 种只在测试集中出现(用黑体标注)。这样设计的目的是检验分类器模型的泛化能力,对未知攻击类型的检测能力是评价入侵检测系统好坏的重要指标。KDD99 数据集总共由 500 万条记录构成,它还提供一个 10% 的训练子集和测试子集。

NSL-KDD 是对 KDD99 的改进数据集,除去了 KDD99 数据集中的冗余数据,克服了分类器偏向于重复出现的记录,学习方法的性能受影响等问题。对正常和异常的数据比例进行了合适选择,测试和训练数据数量更合理,因此更适合在不同的机器学习技术之间进行有效准确的评估。NSL-KDD 分为 KDDTrain+ 和 KDDTest+, 在 KDDTrain+ 中有 125 973 个训练样本,在 KDDTest+ 中有 22 554 个测试样本。

#### 2.2 ISCX2012

ISCX2012 记录的是 7 天时间真实网络环境中的

流量信息,其中包含合法的网络流量以及多种类型的恶意DDoS攻击流量。在这7天的网络流量记录中,DDoS攻击发生时间段为2010/6/14到2010/6/15这两天时间。其中,2010/6/14包含超过960多万条数据报文信息,2010/6/15则包含将近3500万条数据报文信息。

ISCX2012数据集列出了合法或各DDoS攻击类型的数据报文相关信息,包括数据报文类型名,抓包时间,源或目的IP地址,TCP/UDP报文的源或目的端口号等字段信息。

由于2010/6/14和2010/6/15这两天中大部分流量信息都是合法的数据包流量,为了消除数据偏差,在输入到深度学习算法进行训练时,可将所有攻击类型数据报文与随机数目的合法数据报文混合,进行重新采样后再进行输入。

### 2.3 KDN dataset

KDN数据集包含虚拟化网络功能、网络特征、基于深度加强学习的SDN网络路由优化和神经网络四个子数据集,如表1所示。

表1 KDN数据集包含子数据集描述

子数据集名称	所含压缩文件名称	包含数据或源代码描述
虚拟化网络功能	OVS.zip	流量特征和连接到SDN控制器的OVS的CPU消耗
	Firewall.zip	流量特征和配置防火墙规则的OVS的CPU消耗
	Snort.zip	流量特征和经过初始化配置的SNORT的CPU消耗
网络特征	overUnderNetwork.zip	实现负载均衡的overlay-underlay网络中的流量、负载均衡和流层面延时
	star.zip	star网络中的流量和节点对之间的延时
	ring.zip	ring网络中的流量和节点对之间的延时
	scaleFree.zip	无尺度网络中的流量和节点对之间的延时
	overUnder.zip	overlay-underlay网络中的流量和节点对之间的延时
基于深度加强学习的SDN网络路由优化	DRL.zip	深度加强学习代理和OMNeT++网络拓扑的源代码
	train.csv.gz	深度加强学习代理在训练期间的表现
	benchmark.csv.gz	Benchmark数据
	traffic.zip	1000个测试流量矩阵
神经网络	look.csv.gz	100000个测试路由配置
	CodeMatlab.zip	用于训练VNF数据的Matlab代码
	CodePython.zip	用于训练overUnderNetwork数据的Python代码

## 3 快速特征提取

### 3.1 Apache Spark

涉及大量快速数据的应用需要快速处理和大的存储空间,虽然Hadoop系统在提供分布式存储空间方面表现良好,但是在数据处理方面不是太高效,而Spark的提出正是为了解决这一问题。Spark是一个开源的分布式框架,通过对中间结果的内存存储来加快数据处理。Spark可以将Hadoop集群中的应用在内存中的运行速度提升100倍,甚至能够将应用在磁盘上的运行速度提升10倍。Resilient Distributed Dataset (RDD)是Spark的核心,一个RDD代表一个可以被分区的只读数据集,RDD是容错的,Spark提供了多种操作RDD的API。Spark将数据存储为RDD形式,

并可以缓存RDD到内存做进一步处理。

Spark SQL,作为Spark框架的一部分,运行在Spark框架的顶层,主要用于对数据集提供关系型操作。Spark SQL将RDD转变为dataframe,dataframe可以看作关系数据库中的表,允许用户使用sql命令操作相关数据。

### 3.2 HDFS

Hadoop分布式文件系统(HDFS)被设计成适合运行在通用硬件(commodity hardware)上的分布式文件系统,和现有的分布式文件系统有很多共同点,同时区别也是很明显的。HDFS是一个高度容错性的系统,适合部署在廉价的机器上。HDFS能提供高吞吐量的数据访问,非常适合大规模数据集上的应用。HDFS放宽了一部分POSIX约束,来实现流式读取文件系统

数据的目的。HDFS 在最开始是作为 Apache Nutch 搜索引擎项目的基础架构而开发的。

### 3.3 在 KDN 中引入 Spark 和 HDFS

参考文献[11],可以将 Spark 和 HDFS 引入 KDN 架构中的分析平台,实现快速特征提取和实时控制。引入 Spark 和 HDFS 的分析平台如图 3 所示。

其工作流程为:分析平台从控制平面和数据平面收集信息并保存到 CSV 文件,到达一定时间间隔后,将 CSV 文件分布式存储到 HDFS 用于后期处理。在处理阶段,Spark 将 CSV 文件转换成 dataframe,并注册一个临时表,然后 Spark 通过 Spark sql 查询方式提取流量特征。一旦流量特征提取成功,Spark 簇的 master 节点就将这些流量特征发送到分析平台,分析平台再将这些流量特征发送到知识平面供深度学习算法进一步学习,知识平面利用深度学习算法学习到的知识自动地或辅助网络操作员制定具体的网络控制决定。

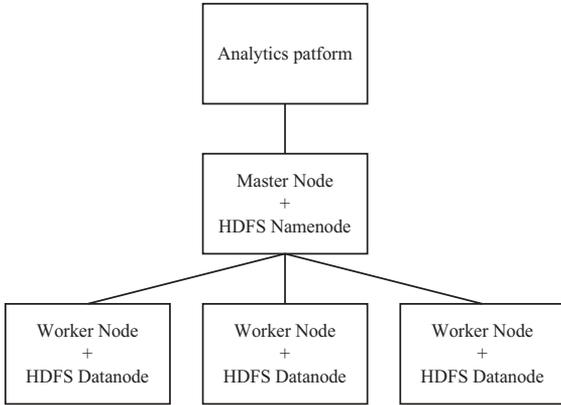


图 3 引入 Spark 和 HDFS 的分析平台

## 4 深度学习算法选择

### 4.1 深度学习算法

深度学习算法可分为监督型深度学习算法、非监督型深度学习算法。

#### (1) 监督型深度学习。

监督型深度学习需要提前给训练数据打标签并进行特征提取。在这种情况下,知识平面可以学习到一个描述网络行为的模型,即一个联系网络参数和网络行为的函数。

#### (2) 非监督型深度学习。

非监督型深度学习采用数据驱动来发现知识,能够发掘所分析数据的内在结构,不需要给样本打标签。在这种情况下,知识平面能发现网络操作员意识不到的数据相关性,比如知识平面可能发现本地天气对链路使用的影响。

### 4.2 算法选择

根据网络操作者是否介入网络决定,KDN 架构的

控制环可以分为开环和闭环,知识平面也对应着两套应用,当知识平面应用使用不同的深度学习算法时,可实现不同的网络功能,如表 2 所示。

表 2 开环和闭环下的可实现应用

控制环	监督型深度学习算法	非监督型深度学习算法
闭环	自动化/优化	改进
开环	验证/评价	推荐或建议

#### (1) 闭环。

· 当应用使用监督型深度学习算法时,首先,知识平面可以代表网络管理员利用所获取的网络模型自动地向网络做决定;其次,可以通过优化技术探究学习到的网络模型获得网络最优配置,所以学习到的网络模型可以用于网络优化。

· 当应用使用非监督型深度学习算法时,因为非监督型深度学习能够发掘所分析数据的内在结构,所以发现的知识可以用于改进网络。

#### (2) 开环。

· 当应用使用监督型深度学习算法时,首先,学习到的网络模型可用于验证,在这种情况下,网络管理员在应用新的配置前可以通过查询模型验证是否可行;其次,因为网络操作者可以微调模型相关参数并获得网络表现评估,所以学习到的网络模型可以用于网络表现评价和分析。

· 当应用使用非监督型深度学习算法时,网络操作者可以参考学习到的知识对网络做决定。

### 4.3 深度学习与强化学习相结合

深度强化学习<sup>[12]</sup>将深度学习的感知能力和强化学习<sup>[13]</sup>的决策能力相结合,可以直接根据输入的数据进行控制,是一种更接近人类思维方式的人工智能方法。深度强化学习下会有一个旨在寻找最优策略的 DRL agent,这个 DRL agent 不断与环境进行交互,向环境发送 action,环境再向 DRL agent 反馈 observation 和 reward。DRL agent 的目标就是寻找一个最优策略使 reward 累加和期望最大。例如在文中环境下,DRL agent 的目标可以是寻找一个最优策略,在该最优策略下实现 SDN 网络的最优配置。

## 5 分析深度学习在 SDN 中的现有应用

文献[14]将深度学习与传统机器学习进行了对比,表明深度学习在网络入侵检测中具有一定的优势。文献[15]使用深度全连接神经网络实现了一个 SDN 网络入侵检测应用,所用数据集为 NSL-KDD dataset,从 41 个特征中选取其中 6 个,即 duration, protocol\_type, src\_bytes, dst\_bytes, count, srv\_count, 含义分别是:连接的持续时间,协议类型,从源到目的的字节数,

从目的到源的字节数,在过去两秒钟连接到同一个主机的连接数,在过去两秒钟连接到同一个服务的连接数。当使用相同的六元特征子集时,其准确率明显高于其他算法,但是该算法模型过于简单,只是一个具有三层隐藏层的深度全连接神经网络,准确率也不是太高,如果适当调整模型或采用更复杂的深度学习算法,准确率一定会显著提高。另外,除了用准确率来评估网络性能,还应该考虑吞吐量等其他因素。

## 6 结束语

论述了深度学习在SDN中的应用研究,具体包括架构、数据源、快速特征提取、深度学习算法选择和分析深度学习在SDN中的现有应用五个方面。SDN的智能化需要深度学习,另外,SDN中有许多突出问题亟待解决,如网络入侵检测、流表冲突检测和SDN的故障自愈管理等,而深度学习必将为这些问题带来新的解决办法。当前,深度学习在SDN中的应用还很少,希望该文有助于深度学习在SDN中应用的进一步推广。

### 参考文献:

- [1] 左青云,陈鸣,赵广松,等. 基于OpenFlow的SDN技术研究[J]. 软件学报,2013,24(5):1078-1097.
- [2] 张顺森,邹复民. 软件定义网络研究综述[J]. 计算机应用研究,2013,30(8):2246-2251.
- [3] HINTON G E, SALAKHUTDINOV R R. Reducing the dimensionality of data with neural networks [J]. Science, 2006,313(5786):504-507.
- [4] CORTES C, VAPNIK V. Support-vector networks[J]. Machine Learning, 1995,20(3):273-297.
- [5] ALTMAN N S. An introduction to kernel and nearest-neighbor nonparametric regression [J]. American Statistician, 1992,46(3):175-185.
- [6] HOSMER D W, LEMESHOW S. Introduction to the logistic regression model[M]. 2nd ed. [s.l.]:[s.n.], 2000:1-30.
- [7] SCHAPIRE R E. The boosting approach to machine learning: an overview [M]//Nonlinear estimation and classification. New York:Springer, 2003:149-171.
- [8] LE Q V. Building high-level features using large scale unsupervised learning [C]//2013 IEEE international conference on acoustics, speech and signal processing. Vancouver, BC, Canada:IEEE, 2013:8595-8598.
- [9] MESTRES A, RODRIGUEZ-NATA A, CARNER J, et al. Knowledge-defined networking [J]. ACM SIGCOMM Computer Communication Review, 2017,47(3):2-10.
- [10] CLARK D D, Partridge C, Ramming J C, et al. A knowledge plane for the internet [C]//Proceedings of the 2003 conference on applications, technologies, architectures, and protocols for computer communications. Karlsruhe, Germany: ACM, 2003:3-10.
- [11] KARIMI A M, NIYAZ Q, SUN Weiqing, et al. Distributed network traffic feature extraction for a real-time IDS [C]//IEEE international conference on electro information technology. Grand Forks, ND, USA:IEEE, 2016:522-526.
- [12] MNIH V, KAVUKCUOGLU K, SILVER D, et al. Human-level control through deep reinforcement learning [J]. Nature, 2015,518(7540):529-533.
- [13] SUTTON R, BARTO A. Introduction to reinforcement learning [J]. Machine Learning, 2005,16(1):285-286.
- [14] DONG Bo, WANG Xue. Comparison deep learning method to traditional methods using for network intrusion detection [C]//IEEE international conference on communication software and networks. Beijing, China:IEEE, 2016:581-585.
- [15] TANG T A, MHAMDI L, MCLERNON D, et al. Deep learning approach for network intrusion detection in software defined networking [C]//International conference on wireless networks and mobile communications. Fez, Morocco: IEEE, 2016:258-263.