Spark 框架下聚类模型在网络流量异常检测中的应用

◆周显春　肖　衡

(三亚学院信息与智能工程学院　海南　572022)

摘要：本文在 Spark 平台上采用基于 RDD 的聚类模型对网络流量异常进行检测。在 Spark 的集群环境下，通过对比测试准确率、WCSS 发现，k-means++聚类模型比 BisectingKMeans 模型更加适合对网络流量进行检测。该实验结果对从事网络流量异常的检测的研究者有一定的借鉴作用。

关键词：网络流量检测；Spark ；k-means++；BisectingKMeans

0 引言

近年来，随着“互联网+”、云平台、大数据等新技术高速发展，互联网现在已经成为经济发展和社会进步的不可或缺的推动力量。与此同时，网络信息的数据量也呈现爆炸式增长，呈现 4V 特性（量大、多样性、速度快、价值密度低）。在大数据的环境下，原有的病毒、黑客、电子窃听、电子欺诈的检测技术效率底下，使得网络的安全问题尤其突出。

同时，为了满足大容量数据分布式处理的要求，国外研究者提出 Apache Spark。Spark 是一种分布式、开源的计算框架，目的是为了简化基于计算机集群的并行程序的编写。Spark 不仅可以发挥 MapReduces 的对大数据的处理能力[1]，还可以充分利用数据集内存缓存、启动任务的低迟延、迭代类运算、实时计算的支持和强大的函数式编程接口[2]。国外学者已经在 Spark 平台用使用机器学习算法 KMM 检测网络流量异常，而且检测效果较好[3]。但是聚类算法在检测网络流量异常检测时，仍然存在对分类数 K 值和初始化中心缺乏有效机制保证的缺陷。针对这个问题，无论是在非 Spark 还是 Spark 平台上，有国内学者提出改进 KMM，实验证明检测效果很好[4-7]，但是在 Spark 平台上研究网络流量异常的较少，尤其是对各种聚类方法检检测效果的对比研究。

因此，本文提出在 Spark 平台上利用各种常见的聚类模型进行网络流量异常检测，对比 k-means++、BisectingKMeans 的测试效果进行比较，找到更适合网络流量异常检测的方法，为有效分析网络海量数据提供一条有力的解决途径。

1 相关研究

1.1 网络流量检测技术的现状

目前，基于机器学习算法的网络流量异常检测方法分为监督学习、无监督学习和半监督学习。其中，有监督学习网络流量异常检测方法首先需要使用监督学习机器学习算法先对带有特征值的训练样本进行训练得到一个预测值，然后把预测值和实际的流量类型进行对比，最后用调试好参数的模型去检测新接受的网络数据，判断其是正常数据，还是异常数据。但需要有已知类型的训练样本[8]，而且是不能检测未知类型的数据，实时性检测效果差。有监督学习网络流量异常检测方法直接对带有特征值的训练数据进行分析，调试好参数，就可以用于检测新的接受网络数据，得出所属类型，检测效果很好[9-10]，但是算法复杂度高[11]。半监督学习把监督学习和无监督学习进行结合，在预测精度和需要已知类型的样本之间取得了很好的折中[12]。

1.2 聚类方法

聚类方法，是数据进行分类，让所有类似的数据在一簇，它是一种无监督的学习方法。常见的聚类算法主要包括 K-均值聚类、模糊 K-均值、层次聚类（凝聚聚类和分列式聚类）等。在聚类方法中，初始质心得选择和质心的数量是找到一个最优模型的关键。如果它们的值初始化或选择不恰当，会造成聚类局部最优，不能找到最有参数，造成预测效果差。因此，研究聚类方法的质心及其数量就成为重点、难点。

Zhang Tian 等[13]提出了一种 Canopy-Kmeans 算法，在 K-均值聚类算法执行之前，先执行 Canopy 算法预处理。后面还有很多学者也展开了相关研究，然而，都并没有从根本上解决初始选值的问题。尤其是随着大数据时代的到来，单一的节点已经无法处理海量数据，需要能够高效、简单的能够在集群上并行运行的算法。

1.3 大数据分布式计算模型

为了应付爆炸式增长数据的有效分析，研究者提出了两种基本的大数据分析开源计算模型：Hadoop 和 Spark。Hadoop，它是受 2003 年至 2006 年 Googl 公司的 GFS、MapReduce、Big Table 的启发而开发了三大神器：满足海量数据访问和存储的分布式文件系统（HDFS）、高效和并行的计算机编程模型（MapReduce）、支持海量数据管理的 Hase，能够满足分布式存储、访问、分析、检索大容量数据的要求，但是对数据科学家来说，存在不能满足数据缓存和支持迭代算法的要求。2009 年，Spark 由加州大学伯克利分校 AMPLab 实验室开发，是对 Hadoop 强有力的补充，可以在 YARN(Hadoop2)所支持的 MapReduce 上运行，还可以与其他开源 Mesos，EC2 平台集成。Spark 是用 Scala 语言实现的，但是开发接口语言不一定是 scala 语言，还可以是 Python、R、Java 语言。Spark 设计理念与核心基于 RDD(Resilient Distributed Dataset) 和采用有向无环图的任务调度机制，可以让中间输出结果可以保存在内存中，从而不再需要读写 HDFS，因此 Spark 能更好地适用于数据挖掘与机器学习等需要迭代的算法[14-16]。

1.4 基于 Spark 的聚类方法

在 Spark 平台上实现的聚类方法有 K-means、GaussianMixture、BisectingKMeans、LDA。其中在网络流量检测方面应用的方法主要有 K-means、BisectingKMeans。K-means 聚类是一种非常经典的挖掘算法，BisectingKMeans 是一种结构化的聚类方法，但是都具有初始值不稳定，容易陷入局部最优的缺点。

1.4.1 k-means++算法流程

（1）首先随机初始化 K 个聚类中心；

（2）计算所有样本距离 K 个聚类中心的欧式距离；

（3）每个样本比较距离 K 个聚类中心的欧式距离，找到最小距离后，将其归纳到距离这个样本最小的聚类中心；

（4）计算每个聚类样本距离该聚类中心距离的平均值，调整聚类中心；

（5）迭代处理（2）~（4），一直到每个聚类中心不再调整，或者该聚类的平均距离小于某个阈值。

1.4.2 BisectingKMeans

（1）所有样本自称为一簇；

（2）然后 WCSS 下降最快的点划分为两个簇；

（3）重复（1）、（2）、（3），一直到簇的数和 K 值相等。

1.5 评价指标

聚类模型性能的评价指标分为内部指标和外部指标。内部指标是以欧式距离、马氏距离等为依据，Spark 平台上实现了欧式

距离，名称为聚类的方差和（WCSS），找到最小值也就知道了最佳模型。外部指标有 F-measure、Rand measure 等，但是需要训练的数据带有标签才能计算。本实验采用 WCSS 对聚类的效果进行评价，然后找到合适的 K 值。

2 实验与结果分析

2.1 实验环境与实验数据

本实验采用虚拟机上安装 ubuntuserver16.10，并安装了必须 Java8、Hadoop2.7、Spark2.10 搭建 Spark 计算集群平台，具体配置见表：

表 1 集群的配置情况

机器名	角色	内存容量	处理器数目
Master	master	4	1
slave1、slave2	salve	3*2	1

本实验数据采用 Kaggle 大赛的用于网络入侵的数据，每条数据记录了信息发送的字节数、登录次数等属性。共有 38 特征，有 489.8 多万个样本[3]。 通过查准率、误差平方对三种不同聚类方法的检测效果进行对比试验。为了让实验结果更加的可信，需要对数据的特征进行规范化。可以利用 computeColumnSummaryStatisticsh 函数统计，信息如下：

（1）平均值

[48.342430463958564,1834.6211752293812,1093.6228137132127，……，0.057659413800050824]

（2）方差值

[523206.01584971714,8.862924680175377E11,4.160409106799707E11,5.716084530911116E-6,……,0.0533507023089224]

从上面显示的结果，无论是平均值还是方差都能够发现了存在离群点，它会影响聚类检测的效果，因此需要归一化处理。

2.2 聚类效果实验

在 spark 平台上运用 k-means++、BisectingKMeans 对数据处理。对比最优的 K 值及其检测效果。

（1）K 的比较（WCSS 的值缩小为原来的万分之一）

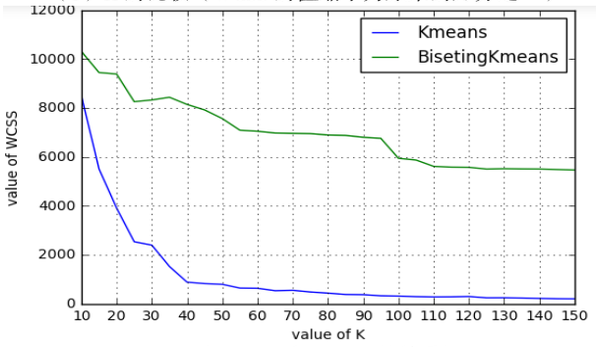

图 1 K 和 WCSS 的关系图

WCSS 的值是选择合适 K 值得依据。它的值越小，聚类的效果就越好。从图 1 可知，Kmeans++算法合适的 K 值是 40，而 BisetingKMeans 算法合适的 K 值是 100。

（2）检测效果比较

用 708M 数据对 KMeans++和 BisetingKMeans 进行训练后，找到最有 K 值，然后预测每个数据是正常数据还是异常数据。把数据带有的标签和预测值相比较，统计效果，包括检测率（检测到的攻击数据占所有攻击数据的比重）和误警率（检测为攻击的正常数据占所有正常数据的比重）。具体检测情况如下表。

表 2 检测效果比较

算法	检测率	误警率
KMeans++	99.94%	24.90%
KMeans++	95.37%	13.2%
BisetingKMeans	93.27	16.54

由表 2 可知，KMean++采用两种不同的区分数据正常和异常的标准。检测率达到了 99.94%，是采用常用的区别标准，直接根据数据所在聚类的类别来划分，简单但是误警率高，达到 24.90%。采用阈值的标准，如果某个点到所属聚类的距离超过了确定的第 110 点距离聚类的距离（先按照距离排序），则判为异常数据。测试效果的误警率降低到 13.2%，但是检测率也下降了。BisetingKMeans 也采用第二种标准。KMean++在 Spark 平台上的检查效果要比 BisetingKMeans 好，误警率较大，达到了 13.2%。而且在实验过程中，BisetingKMeans 训练的时间特别长，不适合在线分析。

3 结束语

本文提出在 Spark 平台上利用聚类模型来进行网络流量异常检测，对比 k-means++、BisectingKMeans 的测试效果进行比较，找到更适合网络流量异常检测的方法，在一定程度上提升了应付网络攻击的能力。但是，本次实验的结果是只分类正常数据和异常数据两类，没有针对攻击类型分类检测，而且采用数据的是离线数据，缺乏实时性，下一步研究的主要方向为：调试聚类参数、修改聚类算法或者借助 SparkStreaming 完成基于聚类的网络流量异常的在线分析等。

参考文献：

[1]唐振坤.基于 Spark 的机器学习平台设计与实现[D].厦门大学，2014.

[2]蔡立宇，黄章帅，周济民译,(南非)Nick Pentreath 著.Spark 机器学习[M].北京：人民邮电出版社，2016.

[3]Sandy Ryza,UrilLaserson,SeanOwen,Josh Wills 著. Spark 高级数据分析.龚少成译[M].北京：人民邮电出版社，2016.

[4]张佃伦.基于粗糙集的聚类算法及其在入侵检测中的应用[D].青岛科技大学，2015.

[5]吴哲夫， 张彤， 肖鹰. 基于 Spark 平台的 K-means 聚类算法改进及并行化实现[J]. 互联网天地， 2016(1):44-50.

[6]李淋淋， 倪建成， 于苹苹.一种基于聚类和 Spark 框架的加权 Slope One 算法[J].计算机应用， 2017.

[7]张波. 基于 Spark 的 K-means 算法的并行化实现与优化[D].华中科技大学， 2015.

[8]陈晓，赵晶玲.大数据处理中混合型聚类算法的研究与实现[J].信息网络安全，2015.

[9]黄俊，韩玲莉，陈光平.基于无指导离群点检测的网络入侵检测技术[J].小型微型计算机系统，2007.

[10]蒋盛益,李庆华.无指导的入侵检测方法[J].计算机工程，2005.

[11]李锦玲，汪斌强.基于最大频繁序列模式挖掘的 App-DDoS 攻击的异常检测[J].电子与信息学报，2013.

[12]陆悠，李伟，罗军舟.一种基于选择性协同学习的网络用户异常行为检测方法[J].计算机学报，2014.

[13]ZHANGT,RAMAKRISHNANR,LIVNYM. BIRCH: an efficient dataclustering method for very large databases[C]// ACM Sigmod Record. 1996.

[14]孙科.基于 Spark 的机器学习应用框架研究与实现[D].上海:上海交通大学，2015.

[15]尹绪森.Spark 与 MLlib: 当机器学习遇见分布式系统[J].程序员，2014.

[16]陈虹君.基于 Spark 框架的聚类算法研究[J].电脑知识与技术，2015.

基金项目：海南省教育科学规划课题成果（QJY13516047）：基于大数据的个性化学习模式构建及实证研究；海南省教育厅科研项目（Hnky2015-55）：面向多媒体的高速率无线传输技术研究；三亚市院地科技合作项目（2015YD11）：基于非连续的宽频谱无线网络传输技术研究。