基于感知哈希矩阵的最近邻入侵检测算法

江泽涛[1],周谭盛子[1],韩立尧[2]
(1. 桂林电子科技大学计算机与信息安全学院,广西桂林 541004;2. 西北工业大学计算机学院,陕西西安 710129)

摘 要: 针对目前入侵检测效率不高的问题,本文提出一种基于感知哈希矩阵的最近邻入侵检测算法. 首先计算训练集中入侵检测对象的感知哈希描述子,并将感知哈希描述子拼接成感知哈希矩阵;然后利用设计好的量化函数对矩阵中的哈希描述子进行量化,并按照感知哈希的性质对矩阵进行约简和调整;在入侵检测阶段用该矩阵快速定位与待检测对象最相近的 K 个样本,利用 K 近邻的投票原则完成入侵检测任务. 通过理论分析及在 KDDCUP99 数据集上的相关实验验证了该方法以 $O(n)$ 的时间复杂度来快速定位最近邻的 K 个样本,在保持高检测率的同时降低了存储和计算方面的开销,从而更加有效的保护网络环境.

关键词: 入侵检测; 感知哈希矩阵; 量化函数; K 近邻; 检测率
中图分类号: TP393.08 **文献标识码:** A **文章编号:** 0372-2112 (2019)07-1538-09
电子学报 URL: http://www.ejournal.org.cn **DOI:** 10.3969/j.issn.0372-2112.2019.07.019

Nearest Neighbor Intrusion Detection Method Based on Perceived Hash Matrix

JIANG Ze-tao[1], ZHOU Tan-sheng-zi[1], HAN Li-yao[2]
(1. *College of Computer and Information Security, Guilin University of Electronic Technology, Guilin, Guangxi* 541004, *China*;
2. *College of Computer Science and Technology, Northwestern Polytechnical University, Xi'an, Shaanxi* 710129, *China*)

Abstract: In view of the low efficiency of current intrusion detection, this paper proposes a Nearest Neighbor Intrusion Detection algorithm based on Perceptual Hash Matrix. Firstly, the perceptual Hash descriptors of the intrusion detection object in the training set is calculated, and the perceptual Hash descriptors are spliced into a perceptual Hash matrix; Then use the designed quantization function to quantize the Hash digest in the matrix, and reduce and adjust the matrix according to the nature of the perceived Hash. In the intrusion detection phase, the matrix is used to quickly locate K samples closest to the object to be detected, using K nearest neighbors(KNN)'s voting principles to complete intrusion detection tasks. Theoretical analysis and related experiments on the KDDCUP99 dataset show that the method can quickly locate the nearest neighbor K samples with the $O(n)$ of time complexity, which can reduce the overhead of storage and calculation while maintaining high detection rate, and more effectively protect the network environment.

Key words: intrusion detection; perceptual Hash matrix; quantization function; KNN; detection rate

1 引言

当今计算机网络成为生活中不可缺少的部分,如何使计算机免受来自互联网的攻击已经成为社会各界密切关注的问题. "入侵检测"自 1980 年被提出后就得到了相关领域的高度关注,这项技术已然成为保障互联网安全的重要防线. 入侵检测系统作为信息安全综合防御系统的重要组成部分已被广泛部署在企事业单位以及公有云和私有云等相关环境中. 如冯子豪[1]提出在工业控制系统中部署 snort 可以检测并预防来自物联网世界的攻击,从而提高了工业控制系统的安全性. Prachi Deshpande[2]等人总结了云环境下的入侵检测系

收稿日期:2018-07-21;修回日期:2019-02-26;责任编辑:李勇锋
基金项目:国家自然科学基金(No. 61572147, No. 61762066, No. 61876049);广西科技计划(No. AC16380108);广西图像图形智能处理重点实验(No. GIIP201701, No. GIIP201801, No. GIIP201802, No. GIIP201803);广西研究生教育创新计划(No. 2018YJCX46);江西省自然科学基金(No. 20171BAB212015)

统,并针对如何将基于主机的入侵检测系统迁移到云环境中提供了相关框架.

大数据与人工智能的到来使得入侵检测的对象更加隐蔽,高速互联网虽然弱化了传统意义下的拒绝服务攻击,但随之而来的却是各种越权攻击和端口攻击,显然传统的入侵检测方法已经不能很好的保护计算机网络的安全. 近年来,深度学习在图像领域取得了革命性的突破,因此也有不少学者将深度网络引入到入侵检测领域中并且取得了很好的效果. 高妮[3]等利用堆叠限制玻尔兹曼机来提取被检测对象的判别特征,然后利用 SVM 在特征空间中构建出超平面来识别不同类型的攻击. 张思聪等[4]等利用深度卷积网络(Deep Convolution Neural Network,DCNN)的特征提取能力,将由属性描述的一维待检测样本转换为二维矩阵并送入深度学习网络中来训练学习,取得了很好的检测效果. 梁杰[5]等人提出利用 GoogleNet 也可以被用来提取待检测样本的特征. 此外,还有利用时间上的先后顺序构建的门控循环单元(Gated Recurrent Unit)和堆叠的 CNN 也可以完成入侵检测[6].

好的深度学习模型不仅取决于网络结构,同时也需要大量的训练样本来调整模型的参数. 但是由于网络流量的不均匀性,深度模型对常见攻击类型比较敏感,而对罕见的攻击往往视而不见. 此外为了在特征空间中找到最优边界需要消耗大量的计算资源,进而在一定程度上限制了深度模型的应用范围.

本文主要从入侵检测数据本身的相关属性出发,提出了一种轻量高效的入侵检测方法并且在公开数据集 KDDCUP99 上做了相关实验,结果表明,本文方法不仅提高了入侵检测的效率,同时可以对罕见攻击进行有效的检测,从而更加有效的保护了互联网安全.

2　基于 KNN 的入侵检测模型

K 近邻(K-Nearest Neighbor,KNN)被广泛应用在入侵检测中,其核心思想是将样本映射到合适的特征空间中,并确定其距离函数就可以完成样本分类. Wagh[7]分析了应用在入侵检测系统中的 KNN 模型并指出基于 KNN 的入侵检测模型不受样本点在特征空间中分布的影响,可以有效避免高维数据中的维度灾难问题. 但是由于在训练阶段没有建立任何学习模型,检测阶段将测试样本与所有的训练样本做相似度计算,将会造成了巨大的空间和计算开销.

为了在训练集合中快速匹配到与检测样本最相似的 K 个样本,Jain[8]等人提出了一种基于快速在线匹配技术(Fast Online Similarity Search,FSS)的 KNN 检测方法. 其基本思想是将样本所在的特征空间递归地划分为若干子空间,然后在最小的特征子空间中匹配最相邻的 K 个样本. 其中 KD-Tree[9]就是基于该思想开发出来的一种快速 KNN 搜索算法,Ball Tree[10]在 KD-Tree 的基础上改进了高维特征空间中遇到的维度灾难问题,从而进一步提高了 KD-Tree 的搜索速度.

本文借鉴了 KD-Tree 特征空间的划分思想,提出了一种感知哈希矩阵(Perceptual Hash Matrix,***PHM***)的检测方法,主要贡献在于提出了感知哈希矩阵的概念,并给出了感知哈希矩阵的构造方法和基于感知哈希矩阵的入侵检测方法. 通过理论分析和相关实验验证了所提方法在提高入侵检测速度的同时可以有效降低计算与存储的开销,从而能够更加有效的保护当前网络环境的安全.

3　基于感知哈希的 KNN 入侵检测模型

3.1　模型设计

基于感知哈希矩阵的入侵检测模型(Perception Hash Based KNN,PH-KNN)的检测流程如图 1 所示.

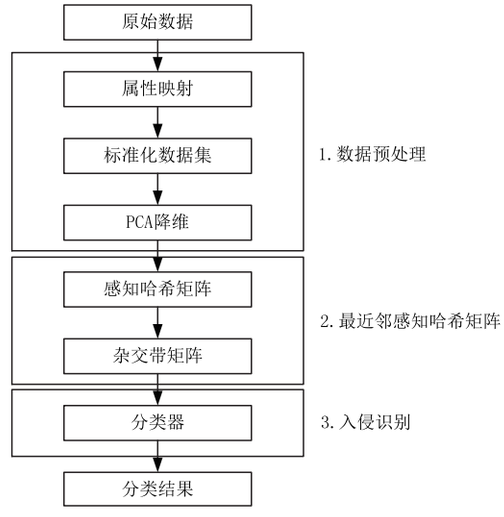

图1　PH-KNN入侵检测模型

该模型主要由三个部分组成,各模块的功能如下:

数据预处理模块:对 KDDCUP99 数据集[11]中的符号变量做数值化处理,然后将待处理的数值变量标准化,最后通过主成分分析(Principle Component Analysis,PCA)降维消除属性之间的相关性.

最近邻感知哈希矩阵构建模块:对经过 PCA 降维后的样本利用感知哈希函数转换为一段感知哈希描述子,并对其进行量化、编码. 感知哈希描述子不仅唯一标识了检测对象,还保留了不同对象之间的相似性关系,由此可用算法 1 来构建感知哈希矩阵. 为了进一步加快入侵检测速度,并用算法 2 构建一

个感知哈希杂交带矩阵(Perceptual Hash Hybridization Matrix,***PHHM***).

入侵识别模块:这一阶段主要完成测试数据的量化和编码工作,同时利用感知哈希矩阵快速定位与待检测样本最近邻的 K 个样本点,最后通过投票选择来完成入侵检测. 具体检测步骤由算法 3 给出.

3.2 PCA 数据降维

PCA 作为一种降维技术,被广泛的应用在数据分析和压缩领域. 其核心思想是对由原始向量构成的矩阵做正交线性变换,从而去掉不同维度上的相关性. 本文利用 PCA 对训练样本做如下变换.

在由观测值 $x_1,x_2,\cdots,x_n$ 构成训练集合中,每个观测值的维度均为 m,则该数据集可用矩阵(1)来表示:

$$\boldsymbol{x}_{m*n}=[x_1,x_2,\cdots,x_n]=\begin{bmatrix}x_{11} & x_{21} & \cdots & x_{n1}\\ x_{12} & x_{22} & \cdots & x_{n2}\\ \vdots & \vdots & & \vdots\\ x_{1m} & x_{2m} & \cdots & x_{nm}\end{bmatrix}=\begin{bmatrix}z_1\\ z_2\\ \vdots\\ z_m\end{bmatrix} \tag{1}$$

式(2)定义了该数据集中不同维度上的均值.

$$\mu=\frac{1}{n}\sum_{i=1}^{m}z_i \tag{2}$$

然后根据式(3)确定某个样本点在不同维度上的偏差.

$$\varphi=\boldsymbol{x}_{m*n}-\mu \tag{3}$$

该数据集协方差矩阵定义为:

$$\boldsymbol{H}=\frac{1}{n}\varphi\varphi^{\mathrm{T}} \tag{4}$$

通过对该协方差矩阵 $\boldsymbol{H}$ 做奇异值分解(Singular Value Decomposition, SVD),可以得到 (λ_1,μ_1),(λ_2,μ_2),$\cdots$,(λ_m,μ_m),这组值是协方差矩阵 $\boldsymbol{H}$ 的 m 组特征值和特征向量,通常将原始数据映射到协方差矩阵中 k 个最大的特征值所对应的特征向量张成的子空间中. 式(5)给出了 k 的确定方法.

$$\sum_{i=1}^{k}\lambda_i\Big/\sum_{i=1}^{m}\lambda_i\geq\beta \tag{5}$$

其中,β 是子空间的特征值之和与原始空间的所有特征值之和的比值. 选取最大的 k 个特征值后,可生成一个大小为 $m*k$ 的矩阵 $\boldsymbol{A}$,按照式(6)将原始数据投影到 k 维子空间中.

$$y=\boldsymbol{A}^{\mathrm{T}}\varphi \tag{6}$$

式(1)~(6)给出了消除数据不同属性间相关性的具体方法,其结果主要用于生成检测对象的感知哈希描述子. 在 KDDCUP99 数据集上调用上述方法生成的特征维度与特征值之间的关系如图 2 所示.

在图 2 中,β 根据经验被设定为 0.854,并由式(5)保留了前 25 个特征向量,这种方法在去掉不同维度相关性的同时可以降低检测过程中的存储开销.

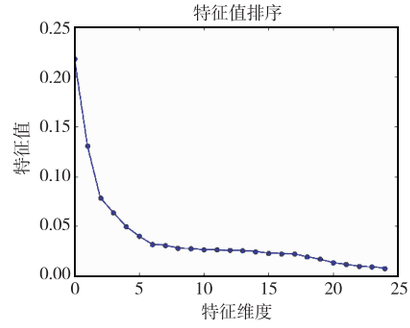

图2 不同主成分对应的特征值

PCA 降维时间复杂度分析:

文献[12]中证明了 PCA 降维的时间复杂度为 $O(\min(n^3,k^3))$,其中,n 表示样本总数,k 表示通过 PCA 降维后得到的子空间维度. 由于 $k<n$,所以 $k^3<<n^3$,因此文章中 PCA 的时间复杂度近似为 $O(k^3)$,相较于样本总数而言,其开销相对较小. 虽然在数据预处理阶段使用 PCA 降维会消耗部分时间,但是降维之后的数据为构建感知哈希矩阵及其杂交带矩阵的构建节省了更多的时间,从而达到了降低时间复杂度的需求.

3.3 感知哈希矩阵的构造

用上一节生成的降维后的样本点按照式(7)计算其感知哈希描述子.

$$Hash(x)=\{[x_1,x_2,\cdots,x_i]\mid x_i=\lfloor x_i*\lambda_i*c\rfloor,0\leq i\leq k\} \tag{7}$$

式(7)生成的感知哈希描述子(感知哈希摘要)由一个 k 维的特征向量组成,即对输入样本点 x 的不同维度分量与该维度分量对应的特征值 λ_i 相乘,并赋予一个加权因子 c,最后对该结果做整数量化得到样本点 x 的感知哈希码. 由式(7)生成的感知哈希描述子满足文献[13]感知哈希摘要的相关性质.

算法 1 感知哈希矩阵构造算法

输入:经过感知哈希函数量化后的入侵检测矩阵

输出:感知哈希矩阵 ***PHM***

步骤:

a)根据感知哈希函数的不同特征描述子确定最大量化阶 N,并为其分配空间指针数组 BUCK[N].

b)将 *Index* 指向最小特征值对应的入侵检测矩阵列.

c)将入侵检测矩阵每一条记录中 *Index* 所指向的值分配到其对应的空间中.

d)把 BUCK[N]中的记录拷贝到原始入侵检测矩阵中,同时清空 BUCK 中的内容.

e)*Index* 指向次小特征值所对应的入侵检测矩阵列,然后返回步骤 *c*).

f)如果 *Index* 已经指向最大特征值所对应的列,结束构造,此时已

经完成了感知哈希矩阵的构造.

g)否则,跳转到步骤 *b*)

算法效率分析:

算法 1 的执行效率与入侵检测矩阵中包含的样本点数目 n,每个样本点的感知哈希描述子个数 k,感知哈希描述子的取值范围 *radix* 有关. 在一次分配与收集中,将所有入侵检测记录分配到 BUCK 中的时间复杂度为 $O(n)$,一趟收集时间复杂度为 $O(radix)$,共进行 k 趟分配和收集. 则算法 1 的时间复杂度为 $O(k*(radix+n))$,同时共需要 $2*radix$ 个指向入侵检测记录的辅助空间.

3.4　基于杂交带的 PH-KNN 算法

图 3 中展示了将训练集合中样本点的感知哈希摘要投影到二维坐标平面上的分布情况.

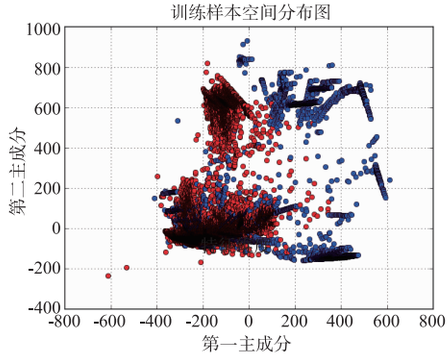

图3　PCA二维映射图

图 3 中的横坐标代表了与第一主成分相对应的坐标分量,纵坐标代表了第二主成分相对应的坐标分量. 红色点表示正常的样本,而蓝色点表示攻击样本(黑色区域是由于样本重叠造成的). 图 3 展示了特征空间中两种类型的样本分布,即单纯型分布—在某确定范围的区域内只存在同种类型的样本点,混合型分布—在某确定范围的区域内存在不同类型的样本点. 同时利用算法 1 构造的感知哈希矩阵样本点的分布情况如图 4 所示,从中可以看出感知哈希矩阵可以很好的描述训练样本点之间的相似近邻关系. 为了提高入侵检测的速度,算法 2 在感知哈希矩阵上构建一个基于混合分布的感知哈希杂交带矩阵(***PHHM***).

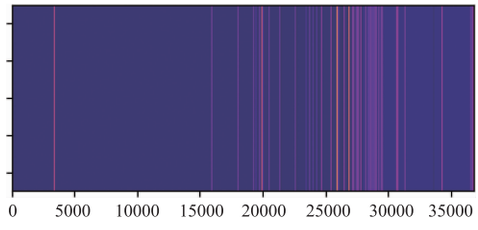

图4　感知哈希矩阵样本分布图

为了计算某个特定区域中正负样本的混杂程度,式(8)定义了杂交系数:

$$coff\_hybrid(x)=\min\left(\frac{|P(x)|}{|P(x)|+|N(x)|},\frac{|N(x)|}{|P(x)|+|N(x)|}\right) \tag{8}$$

$$P(x)=\{1\,|\,x\in \text{positive sample}\} \tag{9}$$

$$N(x)=\{1\,|\,x\in \text{negative sample}\} \tag{10}$$

式(8)中的杂交系数描述了在感知哈希矩阵相邻的区域内不同类别样本之间的混合程度. 该混合程度是区分图 3 中单纯型分布和混合型分布的重要指标. *coff_hybrid* 越小,则该区域内的样本类型越单一,反之样本类型越复杂.

算法 2　基于滑动窗口的感知哈希杂交带矩阵构造算法

输入:感知哈希矩阵 ***PHM***,滑动窗口大小 K,相邻滑动窗口均值之差的阈值 θ,杂交系数阈值 γ.

输出:感知哈希杂交带矩阵 ***PHHM***.

步骤:

a)初始化:

杂交带矩阵 ***PHHM*** = [ ];

滑动窗口的起始位置 start = 0;

前一个滑动窗口的均值:avg_old = start;

当前滑动窗口的均值:avg_new = start;

b)将 ***PHM*** 中包含的罕见攻击添加到 ***PHHM*** 中,同时在 ***PHM*** 消除该类型的攻击;

c)avg_new = avg_new + K;

d)IF avg_new > ***PHM***. length,返回构造的 ***PHHM***;

e)IF abs(avg_old-avg_new) > θ,then avg_old = avg_new,然后跳转到步骤 *c*);

f)IF abs(avg_old-avg_new) < θ,根据式(8)计算当前滑动窗口的杂交系数 coff_hybrid.

IF *coff_hybrid* < γ,then avg_old = avg_new,然后跳转到步骤 *c*);

g)IF *coff_hybrid* > γ,则将当前窗口中包含的感知哈希描述子添加到感知哈希杂交带矩阵中,然后跳转到步骤 *c*).

算法效率分析:

根据样本在杂交带空间的位置关系可以知道,(a)位于杂交带上杂交系数高于非杂交区域;(b)正负样本具有相似的特征属性. 针对步骤 e),如果相邻的滑动窗口位于不同的样本区域,则相邻的滑动窗口中的均值的差值超过给定的阈值 θ,即两个向量滑动窗口的均值大于阈值 θ,那么向下滑动 K 个窗口继续比较. 针对步骤 *f*),如果两窗口的均值之差小于阈值 θ,但是标志数组中的正负样本个数的杂交系数小于阈值 γ,则向下滑动 K 个窗口继续比较. 图 5 给出了按照算法 2 构造的感知哈希杂交带区域上不同类型攻击的分布图,其中不同颜色的线代表不同类型的攻击.

从图 5 中可以看出感知哈希杂交带矩阵中的样本

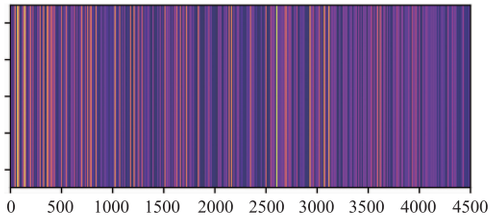

图5 杂交带数据分布图

点远远小于感知哈希矩阵,且包含了多种不同类型的攻击. 实验部分将对生成感知哈希杂交带矩阵的参数做相关分析.

3.5 入侵检测算法

为了计算待分类样本与训练数据之间的相似度,式(11)定义了样本间相似度的计算方法.

$$Dist(T_x,T_s) = \sum \lambda_i (T_{x^i} - T_{s^i}) \tag{11}$$

T_s,T_x 分别代表了训练集合中一条记录的感知哈希码与测试记录的感知哈希码,i 代表当前的维度,λ_i 是第 i 个特征向量对应的特征值. 从式(11)中可以看出如果两条记录的相似度越大,则这两条记录的 *Dist* 越小. 在算法 3 中给出了应用感知哈希矩阵和 *Dist* 距离来做入侵检测的步骤.

算法 3 检测算法

输入:感知哈希矩阵 ***PHM***,感知哈希杂交带矩阵 ***PHHM***,预处理后的待检测样本 S,检测深度 depth.

输出:检测结果.

步骤:

a) 读入预处理的样本 S,调用 PCA 对待检样本做特征降维 SP = PCA(S);

b) 利用式(7)生成 SP 的感知哈希描述子(SPV),即 $SPV = Hash(SP)$;

c) 找到 SP 中最大的前 depth 个特征向量对应的感知特征描述子.

d) IF SPV IN ***PHHM*** THEN 根据式(11)计算距离待测样本 S 最近的 K 个样本 T 的距离 $Dist(SPV,T)$,然后做投票选择.

e) If SPV NOT IN ***PHHM*** THEN 输出 SPV 在 ***PHM*** 矩阵上的判定结果,然后跳转到步骤 *a*).

算法效率分析:

算法 3 利用 ***PHHM*** 和 ***PHM*** 矩阵的样本分布关系可以快速查找到与被检测样本最近邻的 K 个训练样本点. ***PHHM*** 中保存大量的攻击类型,其规模远小于 ***PHM***,适用于异常检测分析;***PHM*** 中保存其余的样本,从而保证了入侵检测的精度. 若 ***PHM*** 的样本点规模为 M,***PHHM*** 的样本点规模为 N,最多需要进行 $O(M+N)$ 次比较就可以确定用来做投票选择的最近邻 K 个样本点. 由于在检测中没有涉及额外的存储空间的分配,因此其空间复杂度为 $O(1)$.

4 实验与分析

4.1 实验设置

4.1.1 实验数据

本文采用入侵检测公开数据集 KDDCUP99 来验证 PH-KNN 算法的有效性,其中每一条记录均是由一条从原始链接中抽取的 41 维特征组成. 其中包含 9 个基本连接特征,13 个内容连接特征和 19 个网络流量特征.

KDDCUP99 的训练集中主要包括四种攻击行为:拒绝服务攻击(Denial of service,DOS)、远程到本地攻击(Remote-to-local,R2L)、越权攻击(User-to-root,U2R)及端口监视和扫描(probing)攻击. 为了与其他相关实验做对比分析,本文将不同的攻击划分为上述四种类型,它们在训练集合中的分布情况如图 6 所示.

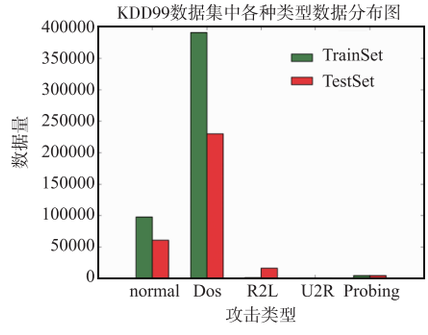

图6 KDD99数据集攻击类型分布

4.1.2 数据预处理

数据预处理主要完成 KDDCUP99 数据集中包含的 3 种符号型属性的数值化,然后对所有特征做标准化处理.

(1)符号型属性数值化

本文将符号型特征转换为数字特征,即对 protocol type 属性中的:'tcp','udp','icmp',分别数值化为 1,2,3. 同理,对'service'属性的 70 种符号和'flag'属性的 11 种符号都建立符号与数值的映射关系.

(2) 特征的标准化

为了消除不同量纲对计算相似度造成的影响,采用 z-scope(式(12))来对数据做标准化处理.

$$new\_data = \frac{ori\_data - ori\_avg}{ori\_std} \tag{12}$$

式(12)中的 *new_data* 代表标准化后的数据,*ori_data* 代表原始数据,*ori_avg* 表示原始数据集合上每一个维度的均值构成的向量,*ori_std* 表示每一个维度上的方差构成的向量.

4.1.3 模型参数设置

本文实验环境为 Dell Precision T700 工作站,主要参数 CPU: Intel(R) Xeon(R) CPU E3-1241 v3 @ 3.5GHz,内存 8GB,操作系统 Windows 8 Home Edition. 使用 Anaconda Python 进行编码实现的. 其中 PCA 降维使用了开源工具 sklearn.decomposition 中的 PCA 库函数[14]. PH-KNN 的参数模型如表 1 所示.

表 1　PH-KNN 模型参数列表

PH-KNN	parameter	Value
PCA	component	25
	copy	true
	whiten	false
	solver	randomized
	iterated_power	5
	β	0.85
算法 1	V. shape	26
	V. count	145586
算法 2	K	52
	θ	15.8
	γ	23
PHM	columns	26
	rows	127907
PHHM	depth	5
	rows	6400
	columns	26

表 1 说明:

本文将原始的 41 维特征向量利用 PCA 投影到 25 维正交特征空间中,在构造 ***PHM*** 和 ***PHHM*** 矩阵中为所有样本添加了相应的类别标签,即 ***PHM*** 和 ***PHHM*** 的 column 为 26. 此外在构造 ***PHM*** 和 ***PHHM*** 矩阵的时候消除了由感知哈希函数生成的重复感知哈希摘要,从而对矩阵的规模进行了压缩. 其中,算法 1 中的参数表示初始时输入样本的大小;算法 2 中的各参数取值均通过网格搜索法产生,根据由经验值设定的参数取值范围,通过组合不同的取值,并以 Accuracy 作为评分标准输出使模型表现性能最好的各参数取值.

4.2 评估标准

为了评估入侵检测算法的性能,本文采用了测试时间(Test Time, TE)、检测率(Detection Rate, DR)、误报率(False Alarm Rate, FAR)、准确率(Accuracy Rate, AC)作为衡量 PH-KNN 模型的性能指标,其定义如下:

$$DR = \frac{TP}{TP + FP} \tag{13}$$

$$FAR = \frac{FP}{TN + FP} \tag{14}$$

$$AC = \frac{TP + TN}{TP + TN + FP + FN} \tag{15}$$

在上述的公式中,TP 表示正确识别的正常记录数,TN 表示正确识别的攻击记录数,FP 表示错误识别的正常记录数,FN 表示错误识别的攻击记录数.

4.3 实验分析

为了验证 PH-KNN 入侵检测模型的有效性,本文设计了两组实验:

实验 1:验证算法 2 中参数对入侵检测效果的影响.

实验 2:比较 PH-KNN 模型与其他入侵检测方法的检测性能.

4.3.1 模型参数分析

(1)验证滑动窗口的大小 K 和两个向量滑动窗口平均值的差值 θ 对生成杂交带矩阵的影响

表 2 给出了 4 种不同大小的 ***PHHM*** 矩阵所对应的 K 和 θ 的相关参数,及其对应的准确率和在测试集合上的检测时间.

表 2　不同参数下杂交带矩阵对分类效果的影响

K	θ	***PHHM***	AC/%	TE/s
30	5	31500x26	95.1674	0.24
40	10	12900x26	97.2667	0.28
50	15	6000x26	98.1196	0.62
60	20	3100x26	87.7603	0.78

从表 2 中可以看出,构建的杂交带矩阵远小于原始矩阵的大小,另一方面杂交带矩阵的大小不仅仅与滑动窗口 K 和相邻滑动窗口差阈值 θ 有关,还与感知哈希矩阵中元素的排布有关.

(2)检测算法中检测粒度对分类结果的影响

图 7 给出了检测粒度与准确率之间的关系,其中,选择了滑动窗口 K 为 50,阈值 θ 为 15 构造的感知哈希矩阵.

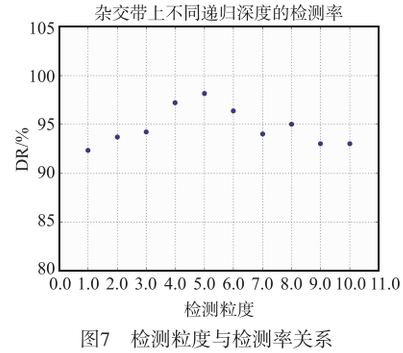

图7　检测粒度与检测率关系

从图7中可以看出,随着检测粒度的增加,检测率呈现先上升,后下降的趋势. 其原因在于增加检测粒度,可以有效避免大数吃小数的现象,有利于对一些罕见的攻击进行检测. 另一方面,检测粒度的增加也会造成过拟合,从而对检测效率造成了负面影响.

4.3.2 KNN 改进算法时间复杂度对比

为了验证本文方法的有效性,分别与不同的 KNN 改进方案在时间复杂度和空间复杂度两个方面进行了相关比较,结果如表3所示.

表3 不同 KNN 算法时间复杂度对比

算法	时间复杂度	空间复杂度
KD-Tree[9]	$O(d\log_2 n)$	$O(n)$
Ball Tree[10]	$O(dn\log n)$	$O(n)$
PH-KNN	$O(\max(n,\min(n^3,d^3)))$	$O(n)$

表3中的 n 表示训练集合中样本点的个数,d 表示每一个样本点的维度,即计算样本之间的距离需要付出的代价. 从表3中可以看出,对 KNN 的不同改进算法主要集中通过改变最近邻样本的搜索策略来降低 KNN 的时间复杂度,而空间复杂度则取决于训练集合上样本点的数量. 与其他的 KNN 改进算法相比,PH-KNN 在时间上消去了由样本维度造成的时间开销,并且通过优化比较策略大大减少了寻找最近邻的 K 个样本所需要的比较次数;空间复杂度上利用感知哈希描述子约简了样本点的数量,从而提高了入侵检测效率.

4.3.3 相关实验对比分析

本实验中,在 KDDCUP99 中的 corrected 数据集上随机抽取10%测试数据集,并按照表1配置了 PH-KNN 的模型参数,其对比的实验结果如表4所示. 表4给出了 PH-KNN 模型与其他方法在检测率、误报率等方面的性能对比.

表4 不同分类器检测性能对比

	Normal		DOS		U2R		R2L		Probe		AC(%)
	DR(%)	FAR(%)	DR(%)	FAR(%)	DR(%)	FAR(%)	DR(%)	FAR(%)	DR(%)	FAR(%)	
ANN-SVM-5[3]	98.8	1.22	98.5	1.18	16.7	79.64	85.9	18.4	95.8	4.77	96.69
DCNN[4]	99.38	1.38	99.49	1.52	Nan	0.0	0.56	0.09	98.79	0.07	99.48
GoogLeNet[5]	98.93	0.33	99.06	1.54	21.08	0.01	0.44	0.19	20.77	2.04	97.27
CNN-GRU[6]	98.37	0.28	98.29	6.64	Nan	0	Nan	0	56.48	0.24	98.15
KD-Tree[9]	98.2	0.31	97.7	1.62	13.2	88.5	64.49	33.78	93.57	6.77	93.71
Ball Tree[10]	99.4	0.61	98.99	0.84	15.48	87.22	74.32	28.43	94.27	6.14	95.16
PH-KNN	**99.5**	**0.12**	**99.1**	**2.84**	**35.5**	**0.01**	**84.06**	**0.27**	**73.4**	**0.27**	**98.34**

从表4中可以看出本文提出的基于 PH-KNN 入侵检测方法具有如下的特点:

1. 与基于最近邻分类的入侵检测方法相比,基于 PH-KNN 的入侵检测模型的检测率略高于其他最近邻入侵检测模型,其主要原因在于 PH-KNN 消除了训练数据中存在的冗余信息对最终分类造成的负面影响,同时由于 PH-KNN 对用于比较的样本做了优化,明显缩短了入侵检测中带来的时间和空间方面的开销.

2. 与基于深度学习的入侵检测方法相比,PH-KNN 模型可以很好的检测到训练集中存在的罕见的攻击类型,从而有效避免了深度学习模型对小样本视而不见的现象. 在检测精度方面,PH-KNN 检测模型的检测精度也略高于其他类型的检测模型.

4.3.4 PH-KNN 对不同类型攻击的检测效果

为了验证 PH-KNN 对不同攻击的检测能力,分别对 KDDCUP99 中39种不同攻击类型进行检测,检测效果如表5所示,其中"/"表示没有当前攻击类型的样本.

从表5中可以看出,PH-KNN 对罕见类型的攻击检测效果显著,原因在于在 PH-KNN 矩阵中保留了所有罕见攻击类型的感知哈希描述子. 而对于正常样本点和 DOS 类型的攻击,PH-KNN 亦能保持较高的检测率.

4.4 PH-KNN 时间复杂度分析

为了突出本文算法在时间复杂度上的优势,特将其与文献[3~6]提出的方法就建模时间和检测时间做了对比实验.

表6中的建模时间表示训练数据集经过数据预处理之后训练模型所需的时间,检测时间表示在测试数据集中随机选取10000个样本,连续操作10次后所得到的平均测试时间. 实验结果表明本文提出的模型在建模时间和测试时间上都具有较大的优势.

PH-KNN 模型的时间复杂度包括数据预处理 $O(\min(n^3,k^3))$,构建的感知哈希矩阵及其杂交带矩阵 $O(k*(radix+n))$,快速定位待检测样本的 K 个最近邻样本 $O(n)$ 三个部分,所以总的时间复杂度为 $O(\max(n,\min(n^3,k^3)))$.

表 5　PH-KNN 对不同类型攻击的检测情况

	类别	训练集合(10%)	测试集合(Corrected)	检测样本数
normal	normal	97278	60593	59623
dos	Back	2203	1098	1027
	Apache2	/	794	/
	Land	21	9	5
	mailbomb	/	5000	/
	neptune	107201	58001	57365
	Pod	264	87	70
	processtable	/	759	/
	smurf	280790	164091	164025
	udpstorm	/	2	/
	teardrop	979	12	5
U2r	buffer_overflow	30	22	11
	httptunnel	/	158	/
	loadmodule	9	2	1
	Perl	3	2	2
	ps	/	16	/
	sqlattack	/	2	/
	xterm	/	13	/
	rootkit	10	13	7
R2l	ftp_write	8	3	0
	guess_passwd	53	4367	4158
	Imap	12	1	1
	Multihop	7	18	7
	named	/	17	/
	Phf	4	2	2
	sendmail	/	17	/
	snmpguess	/	2406	/
	Spy	2	/	/
	warezclient	1020	/	/
	xlock	/	9	/
	xsnoop	/	4	/
	worm	/	2	/
	warezmaster	20	1602	1361
probe	ipsweep	1247	306	301
	mscan	/	1503	/
	nmap	231	84	74
	portsweep	1040	354	288
	saint	/	736	/
	satan	1589	1633	1548

表 6　不同方法运行时间对比

采用方法	建模时间/s	检测时间/s
ANN-SVM-5[3]	289	1.13
DCNN[4]	476	0.785
GoogLeNet[5]	1140	0.46
CNN-GRU[6]	486	0.92
本文	205	0.63

5　结论

本文提出了一种基于感知哈希矩阵的 KNN 的入侵检测模型,该模型利用了感知哈希中相似即相邻的特性设计了基于感知哈希的入侵检测矩阵. 同时提出了利用杂交带矩阵的二级检测方法,降低了 KNN 在入侵检测中在空间存储和计算方面的开销,并在 KDDCUP99 数据集上验证了该模型的有效性. 实验结果表明基于 PH-KNN 入侵检测模型具有较高的检测精度和较低的系统开销,从而可以更加有效的保护网络环境.

参考文献

[1] 冯子豪. Snort 在工业控制系统入侵检测领域的改进及应用[D]. 北京:北京邮电大学,2017.

[2] Prachi Deshpande, S C Sharma, et al. HIDS: A host based intrusion detection system for cloud computing environment [J]. International Journal of System Assurance Engineering and Management, 2018, 9 (3): 567 – 576.

[3] 高妮,高岭,贺毅岳,王海. 基于自动编码网络特征降维的轻量级入侵检测模型[J]. 电子学报,2017,45(3):730 – 739.
GAO N, GAO L, HE YY, WANG H. A lightweight intrusion detection model based on autoencoder network with feature reduction [J]. Acta Electronica Sinica, 2017, 45 (3): 730 – 739. (in Chinese)

[4] 张思聪,谢晓尧,徐洋. 基于 dCNN 的入侵检测方法[J/OL]. 清华大学学报(自然科学版):1 – 9. https://doi.org/10.16511/j.cnki.qhdxxb.2019.22.004. [2019 – 01 – 06].

[5] 梁杰,陈嘉豪,张雪芹,等. 基于独热编码和卷积神经网络的异常检测[J/OL]. 清华大学学报(自然科学版):1 – 7. https://doi.org/10.16511/j.cnki.qhdxxb.2018.25.061. [2019 – 01 – 06].

[6] Chawla A, Lee B, Fallon S, et al. Host based intrusion detection system with combined cnn/rnn model [A]. Proceedings of Second International Workshop on AI in Security[C]. Dublin, Ireland, 2018. 149 – 158.

[7] Wagh S, Neelwarna G, Kolhe S. A Comprehensive Analysis and Study in Intrusion Detection System Using k-NN Algorithm. Multi-disciplinary Trends in Artificial Intelligence

[M]. Berlin Heidelberg: Springer, 2012. 143 – 154.
[8] Jain P, Kulis B, Dhillon IS, Grauman K. Online metric learning and fast similarity search[A]. Proceedings of the 21st International Conference on Neural Information Processing Systems NIPS'08[C]. USA: Curran Associates Inc, 2008. 761 – 768.
[9] Friedman JH, Bentley JL, Finkel RA. An algorithm for finding best matches in logarithmic expected time[J]. ACM Trans Math Softw, 1977, 3(3): 209 – 226.
[10] Liu T, Moore AW, Gray A. Efficient exact K-NN and nonparametric classification in high dimensions[A]. Proceedings of the 16th International Conference on Neural Information Processing Systems[C]. MIT Press, 2003. 265 – 272.
[11] Stolfo S J, Fan W, Lee W K, et al. Cost-Based Modeling for Fraud and Intrusion Detection: Results from the JAM Project[EB/OL]. http://kdd.ics.uci.edu/databases/kddcup99/kddcup99.html, 2011-06-27.
[12] Johnstone I M, Lu A Y. Sparse principal components analysis [J/OL]. https://www.ixueshu.com/document/de76061077a4d849318947a18e7f9386.html, 2004-02-01.
[13] 牛夏牧, 焦玉华. 感知哈希综述[J]. 电子学报, 2008, 36(7): 1405 – 1411.
NIU X M, JIAO Y H. An overview of perception Hashing [J]. Acta Electronica Sinica, 2008, 36(7): 1405 – 1411. (in Chinese)
[14] Tipping M, Bishop C. Probabilistic principal component analysis[J]. Journal of the Royal Statistical Society, Series B, 61, Part 3: 611 – 622.

作者简介

江泽涛 男. 1961 年出生于江西南昌. 博士、教授. 主要研究方向为信息安全、图像处理. E-mail: zetaojiang@126.com

周谭盛子 女. 1993 年出生于安徽宣城. 硕士研究生. 主要研究方向为信息安全.