基于 ICA 算法和三支决策的入侵检测方法

王 帅 黄树成
(江苏科技大学计算机学院 江苏 镇江 212003)

摘 要 网络入侵行为的多样化和智能化,以及网络数据具有特征维数高和非线性可分等特点,导致了网络数据特征提取不充分和模型分类准确率低等问题。为此,提出一种基于独立成分分析(ICA)算法和三支决策(TWD)的入侵检测算法。利用 ICA 算法将网络连接数据基于极大非高斯性进行特征提取,同时将数据从高维特征空间映射到低维特征空间,以此来消除冗余数据,并通过多次的特征提取来构造多粒度的特征空间。对网络行为进行三支决策。建立的模型在 NSL-KDD、CIC-IDS2017 数据集上的实验结果表明其具有更好的特征提取能力和更精确的分类能力。

关键词 ICA 三支决策 特征提取 入侵检测

中图分类号 TP393.08 **文献标志码** A DOI:10.3969/j.issn.1000-386x.2024.07.041

INTRUSION DETECTION METHOD BASED ON ICA ALGORITHM AND THREE-WAY DECISIONS

Wang Shuai Huang Shucheng
(*School of Computer, Jiangsu University of Science and Technology, Zhenjiang 212003, Jiangsu, China*)

Abstract With the diversification and intelligence of network intrusion behaviors, network data has the characteristics of high feature dimensionality and non-linear separability, which leads to insufficient feature extraction and low model classification accuracy in network data. Therefore, an intrusion detection model based on independent component analysis (ICA) and three-way decisions (TWD) is proposed. The characteristics of network connection data were reduced by using ICA algorithm based on maximal non-Gauss property. The data was mapped from high dimensional feature space to low dimensional space to eliminate redundant data. And a multi-granular feature space was constructed through multiple feature extraction. Decisions were made on network behaviors based on three decision-making theories. Experiments were performed on NSL-KDD and CIC-IDS2017 data set. The results show that the proposed model has better feature extraction capability and more accurate classification ability.

Keywords ICA Three-way decisions Feature extraction Intrusion detection

0 引 言

随着网络在人们的日常生活中扮演着日益重要的角色,网络安全越来越重要,而入侵检测是保障网络安全的重要技术之一[1]。通过基本手段检测网络行为,为防御者提供武器,触发针对这些行为的最佳决策计划[2]。

传统的基于机器学习的入侵检测技术在入侵检测中应用十分广泛。常见的有支持向量 SVM、K 近邻算法 KNN 和随机森林算法 RF。上述方法在一定程度上提高了入侵检测的性能。然而,传统的基于机器学习的算法不能自主地学习特征,需要手工构造特征。

于是,有学者研究基于深度学习的入侵检测方法[3],如深度信念网络和深度自编码器。然而,由于神经网络本身具有一定的局限性[4],例如模型调优,并且现有的基于深度学习的方法需要大量的数据,当未标记数据太多时,入侵检测效果并不理想。此外,现有的入侵检测方法都是传统的二分类方法,当数据特征不足或信息不足时,一些数据会被误分类。

收稿日期:2021-01-12。国家自然科学基金项目(61772244)。**王帅**,硕士生,主研领域:模式识别与智能计算。**黄树成**,教授。

为了解决上述问题,本文将 ICA 算法与三支决策理论相结合,建立了入侵检测模型。利用 ICA 获得不同粒度的特征,利用基于三支决策理论的分类器对网络行为进行分类。最后,在 NSL-KDD、CIC-IDS2017 等数据集上评价算法的综合性能。

1　ICA 算法

1.1　概　述

ICA 算法是实现盲源分离[5]的方法之一,就是将数据集中的信息分解为最大独立成分的特征集合。现在也被应用于疾病监测[6]等。快速独立成分分析(FastICA)是常用 ICA 的算法之一[7]。

假如存在一组随机变量 $\boldsymbol{x}=(x_1,x_2,\cdots,x_n)$,它是由一组独立变量 $\boldsymbol{s}=(s_1,s_2,\cdots,s_n)$ 线性混合产生:

$$\boldsymbol{x}=\begin{bmatrix}x_1\\x_2\\\vdots\\x_n\end{bmatrix}=\boldsymbol{A}\begin{bmatrix}s_1\\s_2\\\vdots\\s_n\end{bmatrix} \tag{1}$$

式中:独立变量 $\boldsymbol{s}$ 在混合矩阵 $\boldsymbol{A}$ 的作用下得到了随机变量 $\boldsymbol{x}$。ICA 算法的任务就是从线性混合后的随机量 $\boldsymbol{x}$ 中估计出其最大独立成分。

1.2　数据中心化和白化

去相关性是 ICA 算法数据预处理,主要包含两个步骤:数据中心化和白化,数据中心化也称为数据标准化。

数据中心化[5]的具体做法是对数据中的每个独立值都减去一个值,使其数据各维度的中心都移到零点。

数据白化[5]的目的是减少待估计的参数,大大减少了 ICA 算法的复杂度,具体做法是将去相关后的数据的方差映射到一个范围,一般是使其具有单位方差,即:

$$\begin{aligned}&\boldsymbol{z}=\boldsymbol{w}^{\mathrm{T}}\boldsymbol{x}\\&\text{s.t. }E\{\boldsymbol{z}\boldsymbol{z}^{\mathrm{T}}\}=\boldsymbol{I}\end{aligned} \tag{2}$$

数据白化的过程通常由主成分分析 PCA[8]完成,也可以使用特征值分解的方法。

1.3　估计独立成分

ICA 算法基本估计原理:估计变换矩阵 $\boldsymbol{B}\in\mathbf{R}^{d\times n}$。在保证估计完的样本集中所有变量相互独立的前提下,并且所有分量的线性变化不相关,使用极大似然估计法对 $\boldsymbol{B}$ 进行估计。

在实际应用中一般选用最大化目标函数的算法,于是利用极大非高斯性估计原理[9]构建 ICA 算法,这也就是 FastICA 算法。根据该原理选用近似负熵[10]作为非高斯性的目标函数。计算方法如式(3)所示。

$$J(s)\propto[E\{G(\boldsymbol{s})\}-E\{G(\boldsymbol{v})\}]^2 \tag{3}$$

式中:$\boldsymbol{s}=\boldsymbol{b}^{\mathrm{T}}\boldsymbol{z}$,$\boldsymbol{s}$ 表示变换后的一个独立成分;G 表示非线性函数;$E\{\cdot\}$ 表示求期望;$\boldsymbol{v}$ 表示单位方差均值为零的高斯随机量。常用的 G 有:

$$G(\boldsymbol{s})=\ln(\cosh(\boldsymbol{s})) \tag{4}$$

$$G(\boldsymbol{s})=-\exp(-\boldsymbol{s}^2/2) \tag{5}$$

一般地,近似牛顿法被用来求得式(3)的极值,设函数 $G(\boldsymbol{s})$ 的导数为 $g(\boldsymbol{s})$,$g(\boldsymbol{s})$ 的导数是 $g'(\boldsymbol{s})$,在库恩-塔克条件下式(3)的极值可通过式(6)取得。

$$E\{\boldsymbol{z}g(\boldsymbol{b}^{\mathrm{T}}\boldsymbol{z})\}+\alpha\boldsymbol{b}=0 \tag{6}$$

其中,$\alpha=E\{\boldsymbol{b}^{\mathrm{T}}\boldsymbol{z}g(\boldsymbol{b}^{\mathrm{T}}\boldsymbol{z})\}$,令:

$$H(\boldsymbol{b})=E\{\boldsymbol{z}g(\boldsymbol{b}^{\mathrm{T}}\boldsymbol{z})\}+\alpha\boldsymbol{b} \tag{7}$$

式中:$H(\boldsymbol{b})$ 是式(3)的一阶导数,为了使 $\boldsymbol{b}$ 收敛,需要计算其二阶导数,即 $\partial\boldsymbol{H}/\partial\boldsymbol{b}$:

$$\frac{\partial\boldsymbol{H}}{\partial\boldsymbol{b}}=E\{\boldsymbol{z}\boldsymbol{z}^{\mathrm{T}}g'(\boldsymbol{b}^{\mathrm{T}}\boldsymbol{z})\}+\alpha \tag{8}$$

式中:$E\{\boldsymbol{z}\boldsymbol{z}^{\mathrm{T}}g'(\boldsymbol{b}^{\mathrm{T}}\boldsymbol{z})\}\approx E\{\boldsymbol{z}\boldsymbol{z}^{\mathrm{T}}\}E\{g'(\boldsymbol{b}^{\mathrm{T}}\boldsymbol{z})\}=E\{g'(\boldsymbol{b}^{\mathrm{T}}\boldsymbol{z})\}$,因此 $\boldsymbol{b}$ 迭代公式为:

$$\boldsymbol{b}_{k+1}=\boldsymbol{b}_k-\frac{E\{\boldsymbol{z}g(\boldsymbol{b}_k^{\mathrm{T}}\boldsymbol{z})\}+\alpha\boldsymbol{b}_k}{E\{g'(\boldsymbol{b}_k^{\mathrm{T}}\boldsymbol{z})\}+\alpha} \tag{9}$$

式(9)可以进一步简化,在两边同乘以 $E\{g'(\boldsymbol{b}_k^{\mathrm{T}}\boldsymbol{z})\}+\alpha$,并且不考虑 $\boldsymbol{b}_{k+1}$ 的模,可得到:

$$\boldsymbol{b}_{k+1}=E\{\boldsymbol{z}g(\boldsymbol{b}_k^{\mathrm{T}}\boldsymbol{z})\}-E\{g'(\boldsymbol{b}_k^{\mathrm{T}}\boldsymbol{z})\}\boldsymbol{b}_k \tag{10}$$

由于该算法目的是使 $\boldsymbol{b}^{\mathrm{T}}\boldsymbol{z}$ 具有最大非高斯性,只有 $\boldsymbol{b}$ 标准化后,$\boldsymbol{b}^{\mathrm{T}}\boldsymbol{z}$ 的方差才满足单位化的条件。

$$\boldsymbol{b}_{k+1}=\boldsymbol{b}_{k+1}/\|\boldsymbol{b}_{k+1}\| \tag{11}$$

综上,可以估计出某一个独立成分的转换向量 $\boldsymbol{b}$,同理可以多次估计得到转化矩阵 $\boldsymbol{B}$。考虑到 $E\{\boldsymbol{s}_i\boldsymbol{s}_j\}=E\{(\boldsymbol{b}_i^{\mathrm{T}}\boldsymbol{z})(\boldsymbol{b}_j^{\mathrm{T}}\boldsymbol{z})\}=\boldsymbol{b}_i^{\mathrm{T}}\boldsymbol{b}_j=0$,因此 $\boldsymbol{B}$ 是一个正交矩阵。因此使用式(12)作为迭代规则完成 $\boldsymbol{B}$ 正交化。

$$\boldsymbol{B}_{k+1}=(\boldsymbol{B}_k\boldsymbol{B}_k^{\mathrm{T}})^{-1/2}\boldsymbol{B}_k \tag{12}$$

具体算法步骤如算法 1 所示。

算法 1　FastICA 算法

输入:白化特征集,需要估计的独立成分数量。

输出:独立成分提取之后的特征集。

1. 初始化参数:$\boldsymbol{b}_{i,0}(i=1,2,3,\cdots,d)$,使每一个 $b_{i,0}$ 都具有单位范数;令 $\boldsymbol{B}_0=(\boldsymbol{b}_{1,0},\boldsymbol{b}_{2,0},\boldsymbol{b}_{3,0},\cdots,\boldsymbol{b}_{d0})^{\mathrm{T}}$,$k=0$。

2. **while** $\boldsymbol{B}_k$ 与 $\boldsymbol{B}_{k+1}$ 之间的一范数不收敛于 0:

2.1. 更新 $\boldsymbol{b}_{i,k+1}$:

for $i=1,2,3,\cdots,d$ **do**

$$\boldsymbol{b}_{i,k+1}=E\{\boldsymbol{z}g(\boldsymbol{b}_{i,k}^{\mathrm{T}}\boldsymbol{z})\}-E\{g'(\boldsymbol{b}_{i,k}^{\mathrm{T}}\boldsymbol{z})\}\boldsymbol{b}_{i,k}$$

2.2. 对 $\boldsymbol{B}_{k+1}$ 去相关性:

$$\boldsymbol{B}_{k+1}=(\boldsymbol{b}_{1,k+1},\boldsymbol{b}_{2,k+1},\boldsymbol{b}_{3,k+1},\cdots,\boldsymbol{b}_{d,k+1})^{\mathrm{T}}$$

$$\boldsymbol{B}_{k+1}=(\boldsymbol{B}_k\boldsymbol{B}_k^{\mathrm{T}})^{-1/2}\boldsymbol{B}_k$$

$$k=k+1$$

3. 输出:最后一次迭代得到的转换矩阵 $\boldsymbol{B}\in\mathbf{R}^{d\times n'}$。

得到目的矩阵 $\boldsymbol{B}\in\mathbf{R}^{d\times n}$,顺理成章地能得到经过独立成分提取之后的特征集。

2 算法设计

2.1 三支决策理论

三支决策[11]来源于粗糙集。在二支决策的基础上引入延时决策。当所掌握的信息不够充分,对决策的结果没有十足把握的前提下,为了避免将正常的流量错当成攻击流量的可能,延迟决策是一种可靠的选择。等数据信息补充补全,或者数据特征粒度增加后,再进行明确的决策。因此三支决策降低了因信息不足而盲目决策所造成的风险[12]。

误分类通常会伴随着一定的损失成本,如果在入侵检测的研究中,把一个正常的网络行为错误地归为异常行为可能会造成一些麻烦,但是如果把一个入侵的网络行为错误地分类成正常行为就有可能造成灾难性的后果[13]。

对于一个二分类问题,真实的分类标签可以表示为 P(正)、N(负)即接受和拒绝,可以用一个状态集 $\Omega=\{X,\neg X\}$ 来表示,即用某个数据属于 X 与某个数据不属于 X 来表示一个数据的归属问题。三支决策的决策集可以表示为 $D=\{D_{\mathrm{P}},D_{\mathrm{B}},D_{\mathrm{N}}\}$,分别表示正向决策、边界决策以及负向决策。所有决策的代价损失函数如表 1 所示。记 λ_{PP}、λ_{BP}、λ_{NP}分别表示当前数据属于 X 的时候,采取行动 D_{P}、D_{B}、D_{N}时的损失,λ_{PN}、λ_{BN}、λ_{NN}分别表示当前数据不属于 X 的时候,采取行动 D_{P}、D_{B}、D_{N}时的损失。

表 1 决策的代价损失函数

行动	P	N
D_{P}	λ_{PP}	λ_{PN}
D_{B}	λ_{BP}	λ_{BN}
D_{N}	λ_{NP}	λ_{NN}

假设 $0\leqslant\lambda_{\mathrm{PP}}\leqslant\lambda_{\mathrm{BP}}\leqslant\lambda_{\mathrm{NP}}$,$0\leqslant\lambda_{\mathrm{NN}}\leqslant\lambda_{\mathrm{BN}}\leqslant\lambda_{\mathrm{PN}}$,根据文献[14]的推演证明,可以得到如下两个相关阈值的计算公式:

$$\alpha=\frac{(\lambda_{\mathrm{PN}}-\lambda_{\mathrm{BN}})}{(\lambda_{\mathrm{PN}}-\lambda_{\mathrm{BN}})+(\lambda_{\mathrm{BP}}-\lambda_{\mathrm{PP}})} \tag{13}$$

$$\beta=\frac{(\lambda_{\mathrm{BN}}-\lambda_{\mathrm{NN}})}{(\lambda_{\mathrm{BN}}-\lambda_{\mathrm{NN}})+(\lambda_{\mathrm{NP}}-\lambda_{\mathrm{PP}})} \tag{14}$$

式中:$0\leqslant\beta<\alpha\leqslant1$。可以得到如下三条应用到入侵检测领域的规则:

1)如果 $P(X\mid[x])>\alpha$,则该网络行为被归为正类,即该网络行为是入侵行为。

2)如果 $P(X\mid[x])<\beta$,则该网络行为被归为负类,即该网络行为是正常行为。

3)如果 $\beta\leqslant P(X\mid[x])\leqslant\alpha$,则表示当前信息下,无法对该行为采取任何决策,则该行为需要被划分到边界域以等待进一步的处理。

其中,$[x]$表示样本在属性集下的等价类,$P(X\mid[x])$表示将等价类$[x]$分为 X 的概率,在入侵检测的领域则表示为一个网络行为属于入侵行为的概率。

2.2 入侵检测算法的整体流程

本节构建了基于 ICA 算法和三支决策(ICA-TWD)的入侵检测网络结构,如图 1 所示。提出的入侵检测算法包括数据预处理、特征提取,以及利用三支决策理论进行分类。

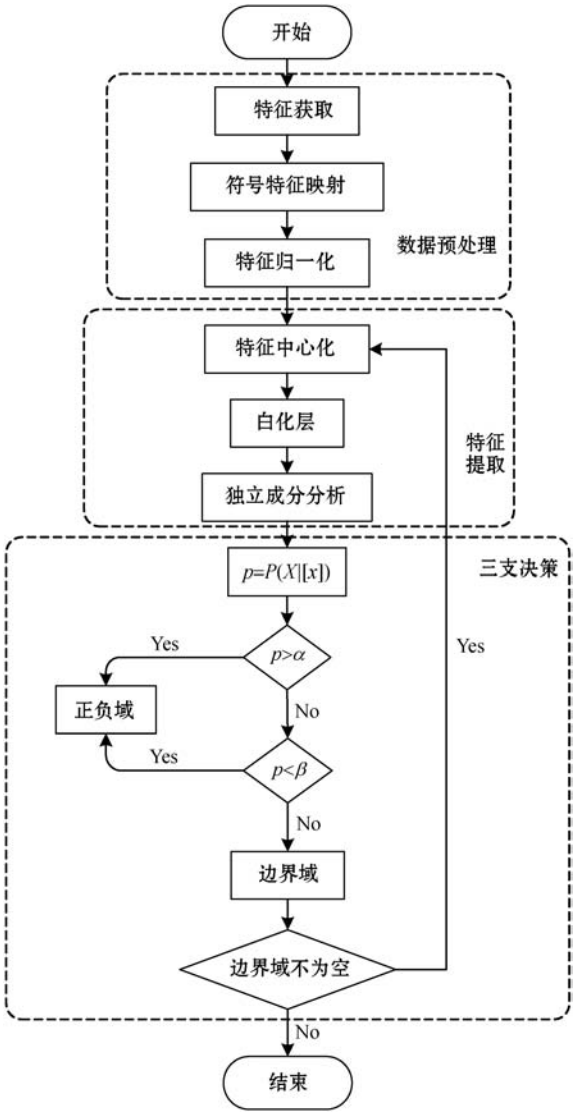

图 1 基于 ICA 和三支决策的入侵检测算法流程

2.3　ICA-TWD 入侵检测算法

在利用三支决策理论进行分类的过程中,根据限制条件,把整个论域分为三个区域(正域、负域和边界域)。在整个决策过程中,都要确定是否对当前的对象做出最终的决策,即确定该对象是属于正域或负域,或应该将不确定的对象归为边界域。

假设样本集为 $X=\{x_1,x_2,\cdots,x_n\}$,样本 x_i 属于正域的概率 $p(P_{OS}\mid x_i)$ 需要被求解出,其中 $i=1,2,\cdots,n$。将 p 值与阈值 α、β 进行比较:若 $p<\beta$,则将其分入负域,若 $p>\alpha$,则将其分入正域,否则分入边界域。

通过对数据中心化和白化处理得到维度较低的数据,然后利用三支决策对数据进行分类,并根据阈值得到分类结果。对于边界域中的数据,在获得额外的信息、重新进行粒度提取后,将被重新评估。在边界域内不再有样本存在之前,这个决策过程将一直持续下去[15]。

具体算法步骤如算法 2 所示。

算法 2　ICA-TWD 算法

输入:训练集 T_r,测试集 T_e。

输出:正域 P_{OS},负域 N_{EG}。

1. 初始化参数:ICA 特征提取方式 G;
   阈值 α,β;
   三支分类器 f;
   正域(P_{OS}) = 负域(N_{EG}) = 边界域(B_{ND}) = $\varnothing$。
2. **while** 测试集 T_e 不为空:
   2.1. $T_r=G(T_r)$;$T_e=G(T_e)$;
   2.2. 根据 T_r 训练模型 f;
   2.3. 由模型 f 得到 T_r 中的每个数据属于正类的概率 $P=f(\tilde{T}_e)$;
   2.4. **For each** $p\in P,t_e\in T_e$:
   If $p>\alpha$:
   $P_{OS}=P_{OS}\cup t_e$
   Else if $p<\beta$:
   $N_{EG}=N_{EG}\cup t_e$
   Else:
   $B_{ND}=B_{ND}\cup t_e$
   End
   End
3. 输出:正域 P_{OS},负域 N_{EG}。

3　实验仿真

3.1　数据集介绍

公开的网络入侵检测数据并不多,本文采用的是 NSL-KDD[16] 入侵检测数据集和 CIC-IDS2017[17] 入侵检测数据集。

3.1.1　NSL-KDD 数据集

NSL-KDD 数据集中每条数据由 41 个特征属性和 1 个类属性组成,其攻击主要分为四种类型:DoS(非法企图中断或干扰主机或网络的正常运行);Probing(非法扫描主机或网络,寻找漏洞、搜索系统配置或网络拓扑);R2L(远程非授权用户非法获得本地主机的用户特权);U2R(本地非授权用户非法获取本地超级用户或管理员的特权)。Normal 表示正常流量。

NSL-KDD 数据集的类型分布如表 2 所示。

表 2　NSL-KDD 数据集分布

类别	Normal	DoS	Probing	R2L	U2R
Train	125 973	45 927	11 656	995	52
Test	9 710	7 458	2 421	2 754	200

3.1.2　CIC-IDS2017 数据集

CIC-IDS2017 数据集包含良性和最新的常见攻击,类似真实世界数据(PCAPs),补充了 NSL-KDD 数据集缺少的各种已知的攻击,比如暴力 FTP、暴力 SSH、渗透、僵尸网络等。

CIC-IDS2017 数据集中每条数据由 79 个特征属性和 1 个类属性组成,数据集分布类型如表 3 所示。

表 3　CIC-IDS2017 数据集分布

时间	标签	数据	合计
周一	BENIGN	529 918	529 918
周二	BENIGN	432 074	445 909
	FTP Patator	7 938	
	SSH Patator	5 897	
周三	BENIGN	440 031	692 703
	DoS GoldenEye	10 293	
	DoS Slowhttptest	5 499	
	Dos Slowloris	5 796	
	Heartbleed	11	
周四上午	BENIGN	168 186	170 366
	Web Attack Brute Force	1 507	
	Web Attack Sql Injection	21	
	Web Attack XSS	652	
周四上午	BENIGN	288 566	288 602
	Infiltration	36	
周五上午	BENIGN	189 067	191 033
	Bot	1 966	
周五下午 1	BENIGN	97 718	225 745
	DDoS	128 027	
周五下午 2	BENIGN	127 537	286 467
	PortScan	158 930	

可以看出,该数据集存在着明显的长尾现象,也就

是数据不平稳问题,数据处理部分针对该问题进行优化。

3.2 数据处理

3.2.1 NSL-KDD 数据处理

1) 离散型符号特征映射。

(1) protocol_type:icmp、tcp、udp 等 3 种。

(2) service:IRC、X11、Z39_50 等 67 种。

(3) flag:OTH、REJ、RSTO 等 11 种。

由于符号与符号之间没有特殊联系,所以就是将离散的符号特征与数字一一对应即可。

2) One-hot 独热编码。

使用 One-hot 编码,将离散特征的取值扩展到了欧氏空间,使得后续计算更加合理[18]。经过编码,将数据集由 41 个特征属性和 1 个类属性扩展成为拥有 119 个特征属性和 1 个类属性的数据集。

3) Min-Max 归一化。

为了使每个特征具有相同的量级,将所有特征进行归一化处理:

$$x^* = \frac{x - x_{\min}}{x_{\max} - x_{\min}} \tag{15}$$

式中:x^* 是归一化后的特征;x 是待归一化特征值;$x_{\min}$ 是该特征的最小值;$x_{\max}$ 是该特征的最大值。在使用所提模型进行分类的过程,可以进行五分类的操作,也可以把五分类转化成五个二分类进行操作,即当 Normal 为正类的时候,其余的样本全部归为负类。本文选取后一种分类操作。

3.2.2 CIC-IDS2017 数据处理

1) 处理重复列。统计中发现数据集中含有重复的两个属性,属性名均为“Fwd Header Length”,任意一个样本的这两个属性值均相等,故判定两个属性为重复属性,所以删除其中一个属性。

2) 处理缺省值。统计发现缺省值存在于“Flow Bytes/s”和“Flow Packets/s”两个属性中,缺省的形式是“infinity”“NaN”或“?”,由于含有缺省值的样本非常少,所以直接删除含有缺省值的样本。

3) 数据不平衡处理。将“Thursday-WorkingHours-Morning-WebAttacks. pcap_ISCX. csv”“Tuesday-Working-Hours. pcap_ISCX. csv”和“Friday-WorkingHours-Morning. pcap_ISCX. csv”三个文件合并成一个文件,因为“Web Attack Brute Force”“Web Attack Sql Injection”和“Web Attack XSS”三种入侵类型样本数较少,将这三种类型合并成一种大的类型,类别标签为“Web Attack”。将三种样本数较少的攻击类型合并成一种类型,可以一定程度上解决数据不平衡带来的问题。

4) Min-Max 归一化。为了使每个特征具有相同的量级,将所有特征进行归一化处理:

$$x^* = \frac{x - x_{\min}}{x_{\max} - x_{\min}} \tag{16}$$

式中:x^* 是归一化后的特征;x 是待归一化特征值;$x_{\min}$ 是该特征的最小值;$x_{\max}$ 是该特征的最大值。

3.3 实验性能评价标准

在入侵检测领域,有两个评判指标比较重要,一个是误报率,一个是漏报率,而漏报率 = 1 - 检出率;精确率反映了被预测为异常的网络行为中有多少是真正的异常行为;F1 分数综合考虑了模型查准率和查全率的计算结果,是反映算法好坏的一个重要指标。

选用准确率 ACC、检出率 DR 和精确率 PR 作为性能评价指标外,外加误报率 FPR 和 F1 得分作为系统性能的评判指标。评价指标计算公式如式(17) - 式(21)所示。

$$A_{CC} = \frac{T_P + N_P}{T_P + F_P + T_N + F_N} \tag{17}$$

$$D_R = \frac{T_P}{T_P + F_N} \tag{18}$$

$$P_R = \frac{T_P}{T_P + F_P} \tag{19}$$

$$F_{PR} = \frac{F_P}{T_N + F_P} \tag{20}$$

$$F_1 = \frac{T_P}{T_P + \frac{P_N + F_P}{2}} \tag{21}$$

式中:T_P 表示入侵流量被正确归类;T_N 表示正常流量被正确归类;F_P 表示正常流量被错归类成入侵流量;F_N 则表示正常流量被错归类到攻击流量。

3.4 样本选取和参数设置

1) NSL-KDD 样本选取。实验在五个不同的样本子集中进行,取五次实验结果的平均值作为实验的性能分析。由于 U2R 类型的攻击相较于其他攻击类型偏少,所以在每个训练集中至少保留 40 条 U2R 类型的攻击数据。同样地,在每个测试集中将保留最少 10 条 U2R 类型的攻击数据用作测试。选取的样本数据子集的类型分布如表 4 所示。其中训练数据集 Train 简写为 Tr,测试数据集 Test 简写为 Te。测试集和训练集之间无重叠。

表 4 五个数据样本子集数据分布

SET	Normal	DoS	Probing	R2L	U2R
Tr0	53 430	36 466	9 301	772	43
Te0	5 349	3 642	912	93	11
Tr1	58 808	40 153	10 159	865	47
Te1	5 846	4 043	1019	88	10

续表 4

SET	Normal	DoS	Probing	R2L	U2R
Tr2	58 840	40 122	10 179	856	48
Te2	5 327	3 640	940	93	12
Tr3	53 474	36 417	9 270	808	41
Te3	5 328	3 691	900	79	14
Tr4	53 578	36 517	9 280	793	42
Te4	5 370	3 724	897	88	10

2）CIC-IDS2017 样本选取。选取的样本数据子集约占总数据集的三分之一,样本类型分布如表 5 所示。

表 5　样本集数据分布

类别	Data set	Train set	Test set
BENIGN	788 818	520 634	26 184
Web Attack	2 180	1 411	769
FTP-Patator	7 935	5 254	2 681
SSH-Patator	5 897	3 935	1 962
Bot	1 956	1 245	711
共计	806 786	532 479	274 307

3）参数设置。本文选择主成分分析(PCA)、奇异值分解(SVD)和因子分析(FA)作为 ICA 的比较方法。

设置主成分分析的超参数为:最大数迭代次数 1 000,最大允许误差 1×10^{-4},线性函数 logcosh,成分数量为 35。设置奇异值分解的超参数为:随机 SVD 求解器的迭代次数 5,成分数量为 35。设置因子分析的超参数为:最大数迭代次数 1 000,最大允许误差 1×10^{-2},迭代次数 3,成分数量为 35。设置 ICA 的超参数为:最大数迭代次数 1 000,最大允许误差 1×10^{-4},线性函数 logcosh,成分数量为 35。

3.5　实验过程及结果分析

1）实验 1。选择主成分分析(PCA)、奇异值分解(SVD)和因子分析(Factor Analysis)验证三支决策分类算法下 ICA 特征提取的可取性。在 NSL-KDD 数据集上进行实验,不同的结果如表 6 所示。

表 6　不同特征提取方法的实验结果对比

模型	ACC	DR	FPR	PR	F1
ICA-TWD	**0.961**	**0.922**	0.033	**0.958**	**0.940**
PCA-TWD	0.937	0.912	0.038	0.939	0.926
SVD-TWD	0.943	0.890	**0.030**	0.935	0.912
FA-TWD	0.929	0.864	0.038	0.918	0.890

可以得出,本文算法 ICA-TWD 具有更高的准确率、F1 评分,检出率、误报率稍稍高于表现最好的 SVD-TWD 模型,综合性能明显优于其他方法。结果表明,将原始数据特征提取到同等维度的条件下,ICA 算法能够保留更多的核心特征。即 ICA 得到的低维特征数据对原始数据的映射效果更好。

图 2 是不同特征提取方法的 ROC 曲线对比,ROC 曲线也被称为感受性曲线,图形曲线可直观地显示方法的准确性,是检验准确度的综合代表之一。其中曲线 F 面积 AUC 可用于评价诊断准确性。

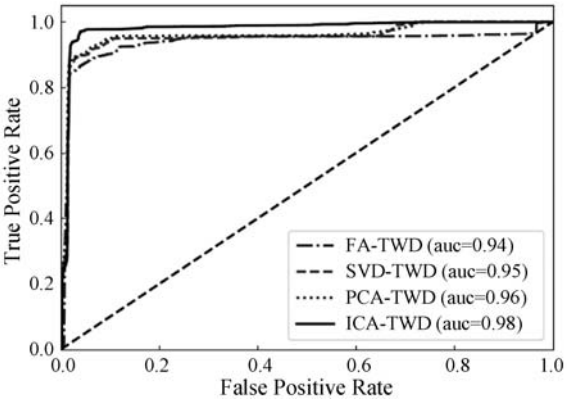

图 2　不同特征提取方法的 ROC 曲线对比

可以看出,ICA-TWD 模型的 AUC 面积最大,证明 ICA-TWD 模型的综合表现更好。

2）实验 2。本文选择支持向量机(SVM)、k 近邻(KNN)、随机森林(RF)和贝叶斯模型(BYS)作为基于三支决策的分类方法的对比模型,此时 ICA 是用于特征提取的方法。在 NSL-KDD 数据集上进行实验,不同结果如表 7 所示。

表 7　不同分类模型的实验结果对比

模型	ACC	DR	FPR	PR	F1
ICA-TWD	**0.961**	**0.923**	0.033	**0.958**	**0.940**
ICA-RF	0.888	0.720	**0.029**	0.925	0.809
ICA-KNN	0.930	0.884	0.047	0.902	0.893
ICA-SVM	0.892	0.790	0.058	0.870	0.838
ICA-BYS	0.855	0.704	0.070	0.831	0.762

可以看出,基于 TWD 的分类模型在准确率(ACC)、检出率(DR)、精度(PR)、F1 分数(F1)四个指标上要优于其他的分类模型得到的结果,尤其是准确率和检出率明显高于其他几种对比模型,这表明通过本文提出的基于三支决策的分类算法在综合性能上优于其他分类算法。在引入延时决策后,避免了一些不确定数据被误分类的风险,大大提高了入侵检测的准确性,把三支决策理论应用在入侵检测上产生了积极的影响,因此得出,基于 TWD 的分类模型更具有优越性。

图 3 是不同方法的 ROC 曲线对比。可以看出,ICA-TWD 模型的 AUC 面积最大,证明 ICA-TWD 模型的综合表现更好。

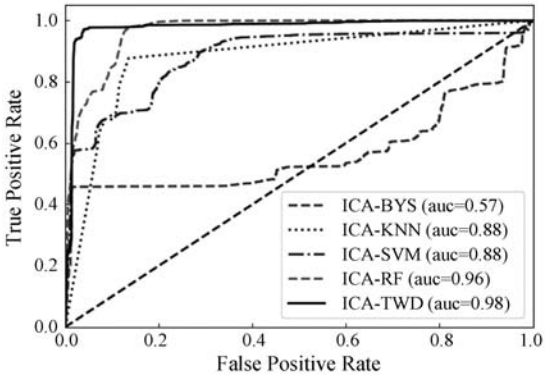

图 3 不同分类模型的 ROC 曲线对比

3）实验 3。本次实验选择的对比模型包括一个基于 LDA 和极限学习机的入侵检测模型(LDA-ELM)[19]，一个基于半监督学习(SSL)的入侵检测模型[20]，一种基于层叠非对称深度自编码器的入侵检测方法(SNADE)[21]和一个基于时空特征的分层入侵检测系统(HAST-IDS)[22]。选取 NSL-KDD 数据集作为实验数据集，表 8 给出了在保持实验环境不变的情况下，本文算法与其他算法的入侵检测对比结果。

表 8 不同算法的实验结果对比(NSL-KDD)

模型	ACC	DR	FPR	PR	F1
ICA-TWD	**0.961**	**0.923**	0.033	0.958	**0.940**
LDA-ELM	0.930	0.898	0.049	0.918	0.881
SNADE	0.925	0.858	**0.031**	0.947	0.902
SSL	0.927	0.907	0.049	**0.965**	0.935
HAST-IDS	0.936	0.922	0.037	0.912	0.903

可以看出，基于 ICA-TWD 的入侵检测模型在准确率(ACC)、检出率(DR)、F1 分数(F1)三个指标上要优于其他的特征提取方法得到的结果，但是在精度(PR)、检出率(FPR)上表现差强人意。综上表明，本文提出的基于三支决策的分类算法在综合性能上优于其他对比模型。

图 4 是不同方法的 ROC 曲线对比。可以看出，ICA-TWD 模型的 AUC 面积最大，证明 ICA-TWD 模型的综合表现更好。

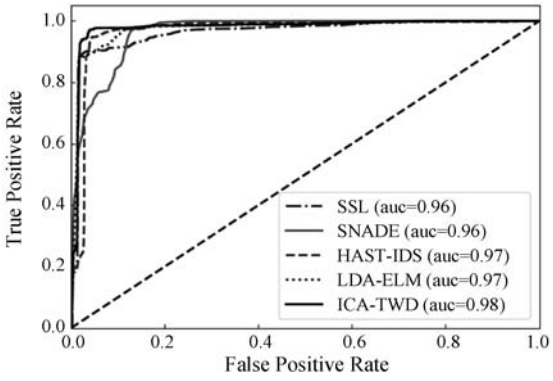

图 4 不同算法的 ROC 曲线对比(NSL-KDD)

从 ROC 曲线上可以看出 ICA-TWD 方法得到的曲线相对于其他方法得到的曲线更接近左上角，且 DAE-TWD 得到的 AUC 面积要大于其他算法得到的 AUC 面积，以上结果表明本文方法的表现要略优于本文所引用的另外四篇文献提出的方法。

4）实验 4。实验 4 是在实验 3 的基础上，将 NSL-KDD 数据集替换成 CIC-IDS2017 数据集，表 9 给出了在保持实验环境不变的情况下，本文算法与其他算法的入侵检测对比结果。

表 9 不同算法的实验结果对比(CIC-IDS2017)

模型	ACC	DR	FPR	PR	F1
ICA-TWD	**0.956**	**0.941**	0.050	0.954	**0.946**
LDA-ELM	0.940	0.911	0.038	0.866	0.905
SNADE	0.942	0.935	0.047	0.882	0.910
SSL	0.950	0.864	**0.037**	0.884	0.871
HAST-IDS	0.925	0.910	0.045	**0.967**	0.934

可以看出，基于 ICA-TWD 的入侵检测模型在准确率(ACC)、检出率(DR)、F1 分数(F1)三个指标上要优于其他的特征提取方法得到的结果，尤其是在指标 F1 分数上比其他最好结果要高出一个百分点，综上表明，在综合性能上，本文提出的入侵检测模型要优于其他对比模型。

图 5 是不同算法的 ROC 曲线对比。可以看出，ICA-TWD 模型的 AUC 面积最大，证明 ICA-TWD 模型的综合表现更好。

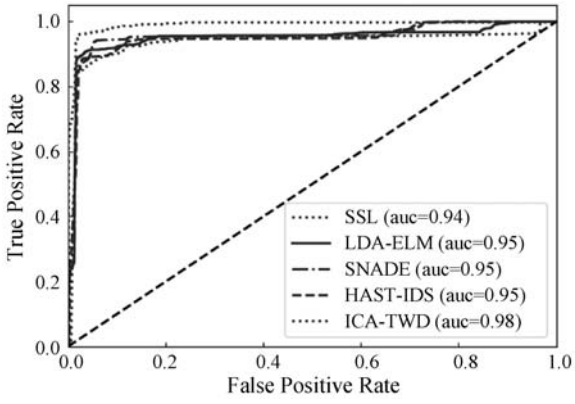

图 5 不同算法的 ROC 曲线对比(CIC-IDS2017)

实验 4 表明本文模型在含有新型攻击类型的入侵检测数据集中仍然具有一定的优越性，说明本文模型具有鲁棒性。

4 结 语

本文提出一种基于 ICA 算法和三支决策的入

侵检测方法。使用 ICA 算法从样本中提取特征,构建多粒度特征空间。利用三支决策理论,根据阈值对网络行为进行分类,将其划分到确定域(正域或负域)中去。对于边界域中的数据,使用基于多粒度特征空间对数据进行重新分类。结合实验,本文算法模型具有更好的特征提取能力和更精确的分类能力。

由于 ICA 算法本身特性,数据特征之间的相关性是我们关心的重点。因此后续研究是通过构建信念网络来分析数据特征之间的独立性,找到一种能够保留更多独立特征的 ICA 算法进行特征映射,进一步提高检测的效率,并构建入侵防御模型,保障网络的安全。

参 考 文 献

[1] Qu F, Zhang J T, Shao Z T, et al. An intrusion detection model based on deep belief network[C]//International Conference on Network, Communication and Computing, 2017: 97 - 101.

[2] 钱燕燕,李永忠,余西亚. 基于多标记与半监督学习的入侵检测方法研究[J]. 计算机科学, 2015, 42(2): 134 - 136, 146.

[3] Tsai C F, Hsu Y F, Lin C Y, et al. Intrusion detection by machine learning: A review[J]. Expert Systems with Applications, 2009, 36(10): 11994 - 12000.

[4] Javaid A, Niyaz Q, Sun W Q, et al. A deep learning approach for network intrusion detection system[C]//9th EAI Endorsed Transactions on Security and Safety, 2016: 21 - 26.

[5] Arabul M U, Rutten M C M, Bruneval P, et al. Unmixing multi-spectral photoacoustic sources in human carotid plaques using non-negative independent component analysis [J]. Photoacoustics, 2019, 15: 100140.

[6] Khanbagi M, Marefat H, Karimi H, et al. Association between integrated cognitive assessment (ICA) and measures of brain structure in mild cognitive impairment and mild Alzheimer's disease[EB/OL]. [2021 - 01 - 02]. https://cognetivity. com/wp-content/uploads/2021/07/43552 _-Association-between-Integrated-C-ognitive-Assessment-ICA-and-Measures-of-Brain-Structure-in-Mild-Cognitive-Impairment-and-Mild-Alzheimers-Disease. pdf.

[7] Safont G, Salazar A, Vergara L, et al. Multichannel dynamic modeling of non-Gaussian mixtures[J]. Pattern Recognition, 2019, 93: 312 - 323.

[8] Gupta A, Barbu A. Parameterized principal component analysis[J]. Pattern Recognition, 2018, 78: 215 - 227.

[9] Malouche Z, Macchi O. Adaptive unsupervised extraction of one component of a linear mixture with a single neuron[J]. IEEE Transactions on Neural Networks, 1998, 9(1): 123 - 138.

[10] Rio L D, Aberg J, Renner R, et al. The thermodynamic meaning of negative entropy[J]. Nature, 2011, 474: 61 - 63.

[11] Nauman M, Azam N, Yao J T. A three-way decision making approach to malware analysis using probabilistic rough sets [J]. Information Sciences, 2016, 374: 193 - 209.

[12] Yao Y. The superiority of three-way decisions in probabilistic rough set models[J]. Information Sciences, 2011, 181(6): 1080 - 1096.

[13] Zhang L B, Li H X, Zhou X Z, et al. Sequential three-way decision based on multi-granular autoencoder features[J]. Information Sciences, 2020, 507: 630 - 643.

[14] 刘盾,梁德翠. 广义三支决策与狭义三支决策[J]. 计算机科学与探索, 2017, 11(3): 502 - 510.

[15] Maldonado S, Peters G, Weber R. Credit scoring using three-way decisions with probabilistic rough sets[J]. Information Sciences, 2020, 507: 700 - 714.

[16] Lakhina S, Joseph S, Verma B. Feature reduction using principal component analysis for effective anomaly-based intrusion detection on NSL-KDD[J]. International Journal of Engineering Science and Technology, 2010, 2(6): 2331 - 2340.

[17] 邓妙然,王开云,张春瑞,等. 网络入侵检测评测数据集对比研究[J]. 现代计算机, 2020(20): 20 - 26.

[18] Matsunaga Y. Accelerating SAT-based Boolean matching for heterogeneous FPGAs using one-hot encoding and CEGAR technique[J]. IEICE Transactions on Fundamentals of Electronics, Communications and Computer Sciences, 2016, 7: 1374 - 1380.

[19] Zheng D H, Hong Z, Wang N, et al. An improved LDA-based ELM classification for intrusion detection algorithm in IoT application[J]. Sensors, 2020, 20(6): 1706.

[20] Li Y Z, Zhang S P, Li Y, et al. Research on intrusion detection algorithm based on deep learning and semi-supervised clustering[J]. International Journal of Cyber Research and Education, 2020, 2(2): 38 - 60.

[21] Shone N, Ngoc T N, Phai V D, et al. A deep learning approach to network intrusion detection[J]. IEEE Transactions on Emerging Topics in Computational Intelligence, 2018, 2(1): 41 - 50.

[22] Wang W, Sheng Y Q, Wang J L, et al. HAST-IDS: Learning hierarchical spatial-temporal features using deep neural networks to improve intrusion detection[J]. IEEE Access, 2018, 6(99): 1792 - 1806.