基于互信息与萤火虫算法的网络入侵特征选择

王新胜　杨　锐
（江苏大学计算机科学与通信工程学院　江苏 镇江 212013）

摘　要　为减少网络入侵检测数据中的冗余特征，提出一种结合互信息和萤火虫算法的特征选择方法。针对互信息不能精确计算特征间冗余度，提出类内特征冗余互信息特征选择方法。针对萤火虫算法步长因子固定易使算法陷入局部最优等问题，提出自适应步长萤火虫算法特征选择。以上方法分别选取特征子集后利用投票策略选取最优子集，对该子集基于 C4.5 和贝叶斯网络分类器分类。实验结果表明，使用 10 个特征检测能有效提高入侵检测率、误报率和 F-measure，同时还缩短训练和检测时间。此外，与现有的几种方法相比，该方法在准确率、检测率和 F-measure 都获得不错效果。

关键词　网络入侵检测　特征选择　投票策略　互信息　萤火虫算法

中图分类号　TP3　　**文献标志码**　A　　DOI：10.3969/j.issn.1000-386x.2024.04.045

NETWORK INTRUSION FEATURE SELECTION BASED ON MUTUAL INFORMATION AND FIREFLY ALGORITHM

Wang Xinsheng　Yang Rui
（*School of Computer Science and Communication Engineering*, *Jiangsu University*, *Zhenjiang 212013*, *Jiangsu*, *China*）

Abstract　In order to reduce redundant features in network intrusion detection data, this paper proposes a feature selection method based on mutual information and firefly algorithm. Aimed at the imprecision calculation of redundancy between features for mutual information, a feature selection method for inner class feature redundancy mutual information was proposed. In order to solve the problem that the fixed step factor in firefly algorithm made the algorithm fall into local optimum, the feature selection of adaptive step size firefly algorithm was proposed. After the feature subset was selected by the above methods, the optimal subset was selected by using the voting strategy. The intrusion detection based on C4.5 and Bayesian network classifier was carried out for this subset. The experimental results show that using 10 features can effectively improve the intrusion detection rate, false alarm rate and F-Measure, and also shorten the training and detection time. In addition, compared with the existing methods, this method achieves good results in accuracy, detection rate and F-Measure.

Keywords　Network intrusion detection　Feature selection　Voting strategy　Mutual information　Firefly algorithm

0　引　言

网络入侵检测系统是网络安全基础设施的重要组成部分，它可以实时抵抗外部攻击，阻止访问内部未经授权的资源文件，能够在攻击造成大面积破坏之前进行检测并及时防护[1]。大部分网络入侵检测系统使用所有特征属性进行检测，然而一些特征存在冗余和无关性，不仅降低入侵检测性能而且还增加了计算复杂度。因此，研究一种有效的网络入侵特征选择算法对网络入侵检测来说具有重大意义。特征选择[2]是网络入侵检测中重要的预处理过程，

收稿日期：2021－01－26。国家自然科学基金项目（61402205）；江苏省自然科学基金青年项目（BK20190838）。**王新胜**，副教授，主研领域：无线传感器网络，网络安全。**杨锐**，硕士生。

不仅可以降低数据维数,还能提高分类准确性。特征选择算法主要分为两大类:过滤法和封装法[3]。过滤法不需依赖特定模型,而是分别计算每个特征与类标签之间关联性,然后根据特征与类标签之间关联性的重要程度进行排序,该方法虽然计算复杂度较小,但适用性较强。在过滤法中,文献[4]提出一种基于信息增益和粗糙集的网络入侵检测方法,该方法首先通过信息增益对数据的各个特征属性进行相关分析,剔除冗余特征属性,然后利用粗糙集理论从数据集中提取分明函数,最后使用随机森林进行分类,有效提高了检测的精度。文献[5]提出一种改进的互信息方法实现网络入侵特征子集优化选择,该方法根据每个特征的互信息评分选取重要特征子集。但以上算法都没有考虑特征组合问题,具有一定局限性。封装法需给出一个预测模型,然后利用该模型对每个特征子集进行评分。根据评分高低,选择最适合模型的特征子集。在封装法中,Natesan 等[6]提出基于封装法的二进制蝙蝠算法进行特征选择,采用朴素贝叶斯分类器实现入侵检测。文献[7]研究了相对其他封装法较优的萤火虫算法解决特征组合优化问题,使用朴素贝叶斯分类器识别网络流量攻击行为。文献[8]使用一种反向学习萤火虫算法来提高网络入侵检测率。虽然文献[6-8]都使用了基于封装法的元启发式算法进行特征选择,且都取得不错的检测效果,但这些算法都容易陷入局部最优等问题。

为解决上述问题,本文综合考虑了这两种方法各自的优势,提出一种 ICFRMI-ASFA 的网络入侵特征选择算法。该算法首先使用类内特征冗余互信息计算特征评分,根据排序结果选出重要特征子集,然后利用自适应步长萤火虫算法选取最优特征子集,最后结合两种方法的特征子集利用投票策略选取最终的特征子集进行入侵检测。

1　特征选择算法 ICFRMI-ASFA

构造有效特征是入侵检测中的重要一步。针对目前数据中特征维数大、影响入侵检测精度等问题,研究人员提出了多种方法进行特征选择。本文为了提高入侵检测性能,提出一种 ICFRMI-ASFA 特征选择方法,所提方法架构如图 1 所示。

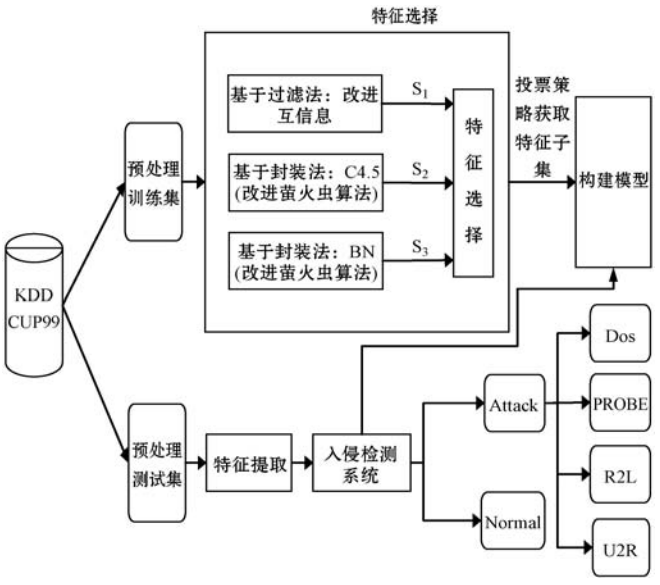

图 1　本文方法架构

1.1　基于类内特征冗余互信息特征选择

互信息是一种有效方式来确定变量之间的相互依赖关系,它可以处理线性相关和非线性相关变量[9],因此本文选择互信息作为过滤法的评价指标。互信息是任意两个随机变量之间关系的对称度量,其值结果非负,如果该值为零说明两个变量之间相互独立。给定两组连续随机变量 $X_i=\{x_1,x_2,\cdots,x_n\}$ 和 $Y_i=\{y_1,y_2,\cdots,y_n\}$,其中 n 为样本总数,它们之间互信息定义如式(1)所示。

$$I(X;Y)=\iint p(x,y)\log_2\frac{p(x,y)}{p(x)p(y)}\mathrm{d}x\mathrm{d}y \tag{1}$$

式中:$p(x)$和$p(y)$分别是 X 和 Y 的边缘概率分布函数;$p(x,y)$是 X 和 Y 的联合概率分布函数;$I(X;Y)$是互信息值,X 是数据集,Y 是类标签。除了用定义的公式直接计算两个变量的互信息外,还可通过计算熵和条件熵来得到变量之间的互信息。其中熵计算如式(2)所示,条件熵计算如式(3)所示,互信息计算如式(4)所示。

$$H(X)=-\int p(x)\log_2 p(x)\mathrm{d}x \tag{2}$$

$$H(X\mid Y)=-\int p(x,y)\log_2 p(x\mid y)\mathrm{d}x\mathrm{d}y \tag{3}$$

$$I(X;Y)=H(X)-H(X\mid Y) \tag{4}$$

式中:$p(x\mid y)$是条件概率,表示在已知 y 发生的条件下 x 发生的概率。对于离散变量来说,通过用求和符号替换定积分形式计算互信息,如式(5)所示。

$$I(X;Y)=\sum_{x\in X,y\in Y}p(x,y)\log_2\frac{p(x,y)}{p(x)p(y)} \tag{5}$$

由于在应用中使用以上公式并不容易从输入数据实例中估计概率密度函数,因此有大量的估计技术来计算互信息。对于估计概率密度函数最常用的方法是直方图和核密度估计[10]。然而,直方图不适用于高维

数据,核密度方法虽然具有很高的估计质量,但计算量较大。因此,对于实验中的高维数据,本文使用 Ross[11] 提出的估计器来计算每个特征与类标签之间的互信息,使用 Kraskov 等[12] 提出的估计器来计算两个连续特征之间的互信息。这两种估计器都通过计算每个点的最近邻的密度来计算互信息,从而避免求解概率密度函数困难,并且这两个估计器使得特征和类标签、特征和特征之间的互信息计算变得相对容易。此外,为了尽可能增大特征与类标签之间的相关性,减少特征之间的冗余度,Ambusaidi 等[13] 提出了如式(6)所示的互信息计算方式。

$$G_{\mathrm{MI}} = \arg\max\left(I(C;f_i) - \frac{1}{|S|}\sum_{f_s\in S}\frac{I(f_i;f_s)}{I(C;f_i)}\right) \tag{6}$$

式中:f_i 是候选特征;f_s 为已选特征;C 是类标签;$|S|$ 为已选特征数量;$I(C;f_i)$ 为候选特征与类标签之间的互信息;$I(f_i;f_s)$ 是候选特征与已选特征之间的互信息。但该方法的特征间冗余度并没有区分类内冗余还是类外冗余,导致冗余度计算不准确。本文为了精确到类内特征冗余,提出如式(7)所示的互信息计算方式。

$$G_{\mathrm{MI}} = \arg\max\left(I(C;f_i) - \frac{1}{|S|}\sum_{f_s\in S}\frac{I(f_i;f_s) - I(f_i;f_s\mid C)}{I(C;f_i)}\right) \tag{7}$$

式中:$I(f_i;f_s\mid C)$ 表示在已知类标签情况下,候选特征与已选特征之间互信息,即类外特征冗余,所以类内冗余为 $I(f_i;f_s) - I(f_i;f_s\mid C)$。因此,基于类内特征冗余互信息的特征选择步骤如下:

(1) 初始化候选特征集合 $F=\{f_1,f_2,\cdots,f_n\}$ 和已选特征集合 $S=\varnothing$。

(2) 计算每个候选特征和类标签之间的 $I(C;f_i)$,找出特征与类标签之间互信息最大的特征,放入已选特征集合 S 中,修改集合 $S=S+f_{\max}$,$F=F-f_{\max}$。

(3) 通过式(7)计算候选特征集合 F 中的每个特征去掉冗余后与类标签的互信息,选出最大互信息特征,然后修改集合 S 和 F,重复操作这一步骤直到候选特征集合 F 为空。

(4) 在已选特征集合 S 中,按照互信息从大到小排序,然后输出已选特征集合 S。

1.2 基于自适应步长萤火虫算法特征选择

萤火虫算法是 Yang[14] 在 2009 年提出的一种元启发式算法,该算法是一种基于种群的智能优化算法,其思想源于自然界中萤火虫个体间的吸引和移动。在萤火虫算法中,有两个极其重要的参数,分别是亮度和吸引力。亮度反映了萤火虫位置的重要性,决定了它运动的方向,吸引力决定萤火虫移动距离。通过萤火虫亮度和吸引力不断更新,实现目标优化。萤火虫亮度定义如式(8)所示。

$$I = I_0 e^{-\gamma r_{ij}^2} \tag{8}$$

式中:I_0 为初始萤火虫亮度;γ 表示光吸收因子;r_{ij} 为萤火虫 i 与萤火虫 j 之间的距离,定义如式(9)所示。

$$r_{ij} = \|x_i - x_j\| = \sqrt{\sum_{d=1}^{D}(x_{id}-x_{jd})^2} \tag{9}$$

式中:d 表示具体维数,D 表示总维数,本文中 D 值为 41。萤火虫之间吸引力定义如式(10)所示。

$$\beta = \beta_0 e^{-\gamma r_{ij}^2} \tag{10}$$

式中:β_0 表示两只萤火虫距离为零时的吸引力,其值一般为 1。当第 i 只萤火虫被第 j 只萤火虫吸引时,萤火虫位置更新公式如式(11)所示。

$$x_i^{t+1} = x_i^t + \beta_0 e^{-\gamma r_{ij}^2}(x_j^t - x_i^t) + \alpha(r_{\mathrm{and}} - 0.5) \tag{11}$$

式中:x_i^t 表示第 i 只萤火虫的当前位置,t 为迭代次数;第二项是萤火虫向周围较亮的萤火虫移动距离,确保算法在更新迭代时靠近最优值;第三项是随机移动,α 为一个常数,其取值范围为 0 ~ 1,r_{and} 是一个随机数生成器,其值均匀分布在 0 和 1 之间。

在上述标准萤火虫算法位置更新公式中,步长因子 α 是一个常数,由于没有考虑算法后期个体间距离随迭代次数的增加而减小,因此固定的步长取值范围容易导致算法产生震荡,具有很大局限性。此外,如果 α 设置较大,算法可能在较小范围里无法尽快收敛;设置较小,可能导致算法在后期迭代过程中陷入局部最优。为了避免上述可能出现的问题,并考虑到萤火虫之间距离远近可通过萤火虫亮度差异体现,亮度差异大说明萤火虫之间距离较远,此时选择大步长,反之选用小步长。因此,本文提出一种自适应步长萤火虫算法,步长公式如式(12)所示。

$$\alpha(t+1) = \frac{I_j(t) - I_i(t)}{I_j(t)}\left(1 - \frac{t}{T_{\max}}\right) \tag{12}$$

式中:$I_j(t)$ 和 $I_i(t)$ 分别为第 i 只萤火虫和第 j 只萤火虫在第 t 次迭代时亮度,并且此时萤火虫 i 飞向萤火虫 j;$T_{\max}$ 表示最大迭代次数。由式(12)可知,在算法初期,步长因子较大有利于全局搜索,随着迭代次数增加,萤火虫之间亮度差异变小,说明萤火虫距离当前最优萤火虫较近,此时,小步长有利于局部搜索。因此,通过自适应动态更新步长因子,从而提高算法收敛速度。当使用萤火虫算法进行特征选择时,萤火虫位置将不再是一个连续值,本文使用离散值 0 和 1 替换,其中 1 表示当前位置特征被选择,反之,0 表示未选择当前位置特征。每只萤火虫的位置可以表示为 $X_i=\{x_1,x_2,\cdots,x_n\}$,$i=1,2,\cdots,n$,其中 n 表示萤火虫数量,$x_{ij}\in\{0,1\}$,d 为特征数量。萤火虫每次迭代位置更新后,新位置值不再

是 0 或 1,为了使算法迭代后萤火虫位置值继续保持二元离散,本文使用式(13)进行位置更新策略。

$$x_{id}^{t+1}=\begin{cases}1 & p_{id}>r_{\text{and}}\\ 0 & \text{其他}\end{cases} \tag{13}$$

式中:$p_{id}=\dfrac{1}{1+\mathrm{e}^{x_{id}^{t}}}$,这里的 x_{id}^{t+1} 表示当前第 i 只萤火虫的第 d 个特征,x_{id}^{t} 表示第 i 只萤火虫先前的第 d 个特征;r_{and} 为一个随机数生成器,其值范围为 0 ~ 1 之间。萤火虫算法作为一种基于封装法的特征选择方法,使用分类器的准确性对所选特征子集进行评价找到最优子集。在本文中使用 C4.5 和贝叶斯网络分类器的准确性作为萤火虫算法的目标函数,准确性低的萤火虫(θ_i)向准确性高的萤火虫(θ_j)方向移动,改进后的移动公式如式(14)所示。

$$x_i^{t+1}=x_i^t+\beta_0\mathrm{e}^{-\gamma r_{ij}^2}(x_j^t-x_i^t)+\alpha(t)(r_{\text{and}}-0.5) \tag{14}$$

由于萤火虫算法选取的特征数量不固定,对后期网络入侵检测模型性能的稳定性产生负面影响,因此,本文为了得到固定特征数量,提出一种自适应策略(AS)。首先固定选取特征数量 w,如果自适应步长萤火虫算法选取的特征数量 m 少于 w,根据未选取特征的互信息大小,将剩余的 $w-m$ 个特征按照互信息从大到小加入到该特征集合中,反之,当 m 大于 w 时,去掉互信息最低的 $m-w$ 个特征。基于自适应步长萤火虫算法的特征选择流程如图 2 所示。

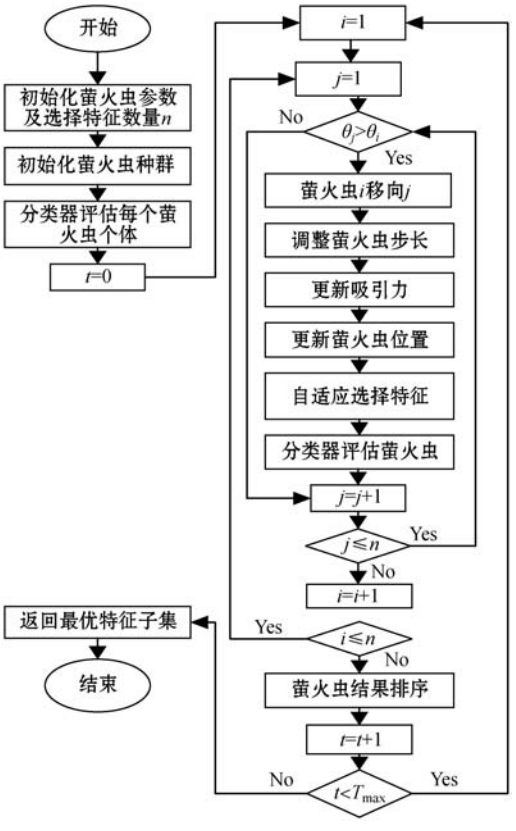

图 2　基于改进萤火虫算法特征选择流程

本文基于 ICFRMIMI-ASFA 特征选择策略如下:使用类内特征冗余互信息方法获取的特征集合为 S_1,使用自适应步长萤火虫算法与 C4.5 分类器和贝叶斯网络分类器选取的特征集合为 S_2 和 S_3。本文根据投票策略选取最终特征子集,投票策略为从三个集合中选取至少同时出现在两个集合中的特征,如式(15)所示。

$$S=\{f:f\in((S_1\cap S_2)\cup(S_1\cap S_3)\cup(S_2\cap S_3))\} \tag{15}$$

2　实　验

2.1　数据集介绍及预处理

本文使用 KDD CUP99 数据集来评估本文算法性能。该数据集是目前最流行的入侵检测数据集之一,被广泛用于评估入侵检测系统性能。该数据集由 5 种类别组成,分别为 Normal、DoS、Probe、R2L 和 U2R,后四种攻击类型又可细分为 22 种小型攻击。该数据集的每条记录包含 41 个属性,其中:前 9 个属性表示基本连接特征,后 13 个属性表示数据包内容特征,另外 9 个属性表示基于时间网络流量统计特征,最后 10 个属性表示基于主机网络流量统计特征。数据集攻击类型详细描述如表 1 所示。

表 1　KDD CUP99 数据集攻击类型

攻击类型	攻击描述
DoS	攻击者耗尽计算资源,拒绝合法用户使用服务
Probe	攻击者试图获取本地超级用户特权
R2L	攻击者利用目标服务器漏洞访问普通用户
U2R	攻击者收集目标主机信息,试图避免安全管理

此外,由于该数据集攻击样本分布不均衡,为了避免数据不均衡对特征选择算法性能产生影响,本文根据样本攻击类别分布来获得训练集和测试集,如表 2 所示。

表 2　实验数据集描述

攻击类别	类型	训练集数量	测试集数量
DoS	Normal	40 000	40 000
	smurf	10 000	10 000
	neptune	5 000	5 000
	back	1 000	1 203
	land	10	11
	pod	100	164
	teardrop	400	579
	总计	56 510	56 957

续表 2

攻击类别	类型	训练集数量	测试集数量
Probe	Normal	40 000	40 000
	satan	800	789
	portsweep	500	540
	nmap	110	121
	ipsweep	600	647
	总计	42 010	42 097
R2L	Normal	40 000	40 000
	ftp_write	4	4
	guess_passwd	23	30
	imap	7	5
	multihop	3	4
	warezclient	520	500
	warezmaster	10	10
	phf	4	0
	spy	2	0
	总计	40 573	40 553
U2R	Normal	40 000	40 000
	buffer_overflow	15	15
	rootkit	4	6
	loadmodule	4	5
	Perl	0	3
	总计	40 023	40 029

KDD CUP99 数据集预处理主要包括数据转换和归一化。由于数据集存在一些协议类型等非数字特征,不利于分类器模型数据输入训练,因此需要进行转换成数字特征。例如 Protocal_type 这个特征属性有 3 种类型,分别为 Tcp、Icmp 和 Udp,需要转换成数字类型,可用 0 ~2 分别代替。数据转换完成后,还需进一步归一化处理,使其每个属性值落在[0,1],计算方法如式(16)所示。

$$x_{\text{norm}} = \frac{X - X_{\min}}{X_{\max} - X_{\min}} \tag{16}$$

式中:$X_{\max}$表示特征的最大值;$X_{\min}$表示特征的最小值。

2.2 实验环境与评估指标

本文所有实验均在硬件环境为 Intel Core i5 CPU 2.5 GHz,8 GB 内存的 64 位 Windows 操作系统上完成。本文关于自适应步长萤火虫算法参数设置如下:吸引力 $\beta_0 = 1$,光吸收因子 $\gamma = 1$,初始步长因子 $\alpha_0 = 0.5$,萤火虫种群数量 $n = 20$,最大迭代次数为 200。

此外,评估 ICFRMI-ASFA 算法的性能指标为准确率(ACC)、检测率(DR)、误报率(FAR)和 F-measure。其中每种指标详细的定义如下。

$$A_{\text{cc}} = \frac{T_{\text{P}} + T_{\text{N}}}{T_{\text{P}} + T_{\text{N}} + F_{\text{P}} + F_{\text{N}}} \tag{17}$$

$$D_{\text{R}} = R_{\text{ecall}} = \frac{T_{\text{P}}}{T_{\text{P}} + F_{\text{N}}} \tag{18}$$

$$F_{\text{AR}} = \frac{F_{\text{P}}}{T_{\text{N}} + F_{\text{P}}} \tag{19}$$

$$P_{\text{recision}} = \frac{T_{\text{P}}}{T_{\text{P}} + F_{\text{P}}} \tag{20}$$

$$F_{\text{-measure}} = \frac{2 \times R_{\text{ecall}} \times P_{\text{recision}}}{R_{\text{ecall}} + P_{\text{recision}}} \tag{21}$$

式中:T_{P} 是将攻击样本正确分类为攻击样本的数量;T_{N} 是将正常样本正确分类为正常样本的数量;F_{P} 是将正常样本错误分类为攻击样本的数量;F_{N} 是将攻击样本错误分类为正常样本的数量。

2.3 实验结果与分析

基于类内特征冗余互信息和自适应步长萤火虫算法选取重要特征结果如表 3 所示。

表 3 不同方法检测四类攻击的重要特征

攻击类别	方法	所选重要特征
DOS	MI	$f_{41}, f_{40}, f_3, f_{10}, f_5, f_6, f_{23}, f_{37}, f_{24}, f_{27}$
	Wrapper(C4.5)	$f_2, f_3, f_5, f_6, f_{10}, f_{12}, f_{23}, f_{24}, f_{27}, f_{41}$
	Wrapper(BN)	$f_3, f_5, f_{12}, f_{22}, f_{23}, f_{25}, f_{27}, f_{31}, f_{34}, f_{40}$
PROBE	MI	$f_{41}, f_{28}, f_{27}, f_{40}, f_5, f_6, f_{39}, f_4, f_{35}, f_3$
	Wrapper(C4.5)	$f_1, f_2, f_3, f_5, f_{10}, f_{28}, f_{31}, f_{39}, f_{40}, f_{41}$
	Wrapper(BN)	$f_2, f_5, f_6, f_{19}, f_{22}, f_{26}, f_{27}, f_{29}, f_{31}, f_{35}$
R2L	MI	$f_{41}, f_{40}, f_1, f_{28}, f_3, f_{33}, f_5, f_6, f_{12}, f_{24}$
	Wrapper(C4.5)	$f_5, f_6, f_7, f_{12}, f_{16}, f_{18}, f_{22}, f_{25}, f_{28}, f_{33}$
	Wrapper(BN)	$f_1, f_5, f_6, f_7, f_{13}, f_{15}, f_{22}, f_{24}f_{25}, f_{31}$
U2R	MI	$f_{41}, f_{33}, f_{28}, f_{40}, f_{10}, f_3, f_5, f_6, f_{34}, f_{36}$
	Wrapper(C4.5)	$f_1, f_3, f_5, f_8, f_{13}, f_{14}, f_{15}, f_{16}, f_{25}, f_{40}$
	Wrapper(BN)	$f_1, f_5, f_6, f_{10}, f_{11}, f_{15}, f_{20}, f_{25}, f_{33}, f_{41}$

表 3 中加下划线的特征表示至少有两种方法选择该特征,可以看出特征 1(网络连接期间的时间间隔)是检测 R2L 和 U2R 攻击的重要特征,因为 R2L 和 U2R 攻击通常需要较长时间来猜测密码或利用某些系统漏洞来登录。此外,不同的特征选择算法会产生相同的重要特征,例如特征 3 是检测 DoS 攻击的重要特征。通过投票策略选取重要的 10 个特征来检测不同类型的攻击,结果如表 4 所示。

表 4 本文方法所选重要特征

攻击类别	所选重要特征
DoS	$f_3, f_5, f_6, f_{10}, f_{12}, f_{23}, f_{24}, f_{27}, f_{40}, f_{41}$
Probe	$f_2, f_3, f_5, f_6, f_{27}, f_{28}, f_{31}, f_{35}, f_{40}, f_{41}$
R2L	$f_1, f_5, f_6, f_7, f_{12}, f_{22}, f_{24}, f_{25}, f_{28}, f_{33}$
U2R	$f_1, f_3, f_5, f_6, f_{10}, f_{15}, f_{25}, f_{33}, f_{40}, f_{41}$

本文采用同样测试集,利用表 4 选取每类攻击的 10 个重要特征和原始 41 个特征进行对比实验,从检测率、误报率及 F-measure 这三个方面比较分析,结果如表 5 所示,其中加下划线数据表示 10 个特征的检测性能优于 41 个特征。可以看出两种方法对 Dos、Probe 和 R2L 都达到较满意的检测结果,而对 U2R 的攻击检测并不是十分有效,这主要由于 U2R 攻击行为与正常活动非常相似。此外,本文还对 C4.5 和贝叶斯网络分类器所选 10 个重要特征和其他特征个数的训练时间进行比较,结果如图 3 和图 4 所示,表明 10 个重要特征构建模型所花费时间要少。另外,C4.5 和贝叶斯网络分类器对所选 10 个重要特征和其他特征个数的测试时间也进行了比较,结果如图 5 和图 6 所示,表明使用 10 个重要特征检测攻击所花费的时间较少。为了体现本文的特征选择方法在模型检测时间上的优势,本文还与文献[13,15]提出的特征选择算法进行对比,结果如图 7 所示。可看出,本文特征选择算法模型检测时间比其他特征选择算法更少,检测速度更快,可以更好满足网络入侵检测要求。

表 5　10 特征与 41 特征检测性能比较

攻击类别	方法	41 特征			10 特征		
		DR/%	FPR/%	F-measure	DR/%	FPR/%	F-measure
DOS	BN	99.78	0.08	0.92	99.97	0.08	0.98
	C4.5	99.95	0.15	0.97	99.98	0.13	0.97
probe	BN	87.74	6.08	0.60	93.45	0.04	0.92
	C4.5	63.04	0.04	0.52	64.53	0.02	0.84
R2L	BN	99.90	0.33	0.85	90.95	0.31	0.84
	C4.5	92.95	0.02	0.88	89.96	0.01	0.94
U2R	BN	75.86	0.29	0.26	68.99	0.14	0.48
	C4.5	31.03	0.00	0.38	17.56	0.00	0.26

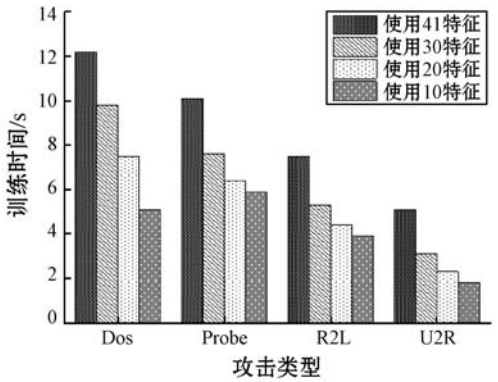

图 3　C4.5 分类器用于选择不同特征个数训练时间比较

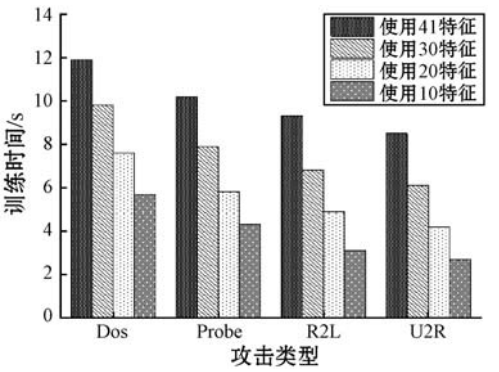

图 4　贝叶斯网络分类器用于选择不同特征个数训练时间比较

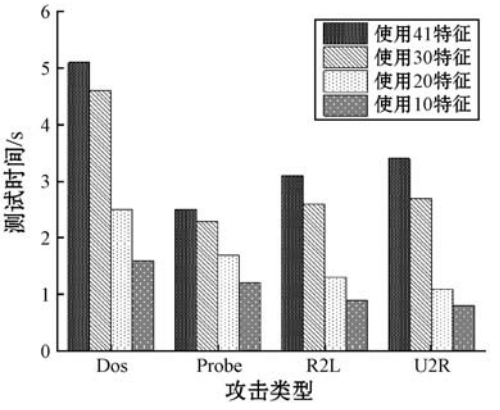

图 5　C4.5 分类器用于选择不同特征个数测试时间比较

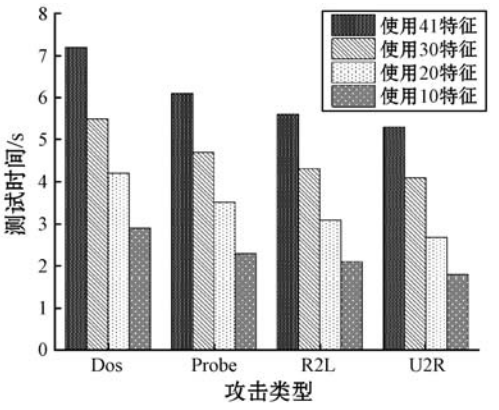

图 6　贝叶斯网络分类器用于选择不同特征个数测试时间比较

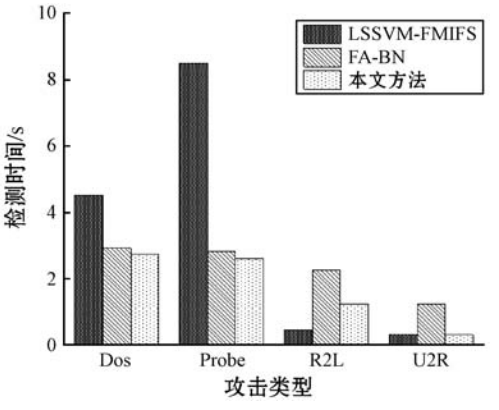

图 7　不同特征选择算法模型检测时间比较

为了验证所提特征选择算法的有效性,使用本文所提的特征选择方法,并以 C4.5 作为分类器与现有的

几种特征选择方法的准确率做了对比分析,结果如表6所示。

表6 本文与其他特征选择算法准确率比较

特征选择算法	特征数量	准确率/%
FGMMI[16]	15	98.01
BFA[7]	14	92.02
FA-SVM[17]	5	78.89
MI-BGSA[18]	5	88.36
本文方法	10	99.23

可以看出,相对于其他互信息和萤火虫算法,本文的特征选择方法在准确率方面有不错的效果,这主要由于本文中互信息考虑了类内特征冗余以及采用自适应步长萤火虫算法,避免算法陷入局部最优,提高入侵检测准确率。

此外,为了进一步体现本文算法的优越性,我们与文献[19]提出的RDD-IDS模型和文献[20]提出的WOAR-SVM模型从检测率和F-measure方面进行比较,结果如图8和图9所示。

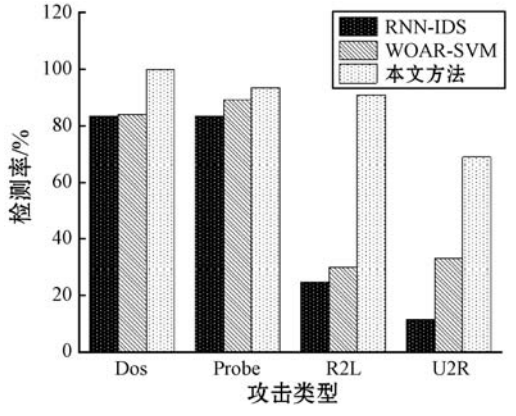

图8 不同算法的检测率比较

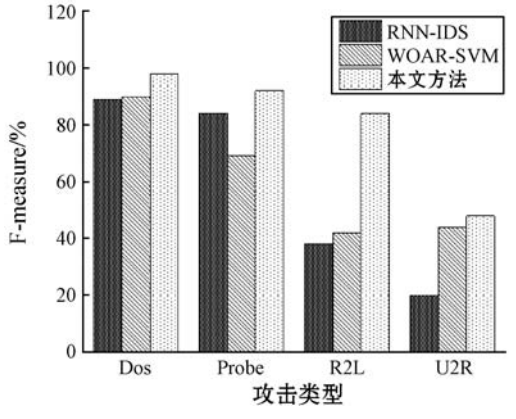

图9 不同算法的F-measure比较

可以看出,本文算法的检测率和F-measure都优于文献[19]和文献[20]方法。这说明本文方法具有较好的入侵检测效果,能够为入侵检测提供一个新思路。

3 结 语

针对当前网络入侵检测数据集中存在较多冗余和不相关特征,导致入侵系统性能不高的问题,本文提出一种基于类内特征冗余互信息和自适应步长萤火虫算法的网络入侵特征选择方法,实验结果表明,选取最优10个特征子集的检测率、误报率和F-measure相比41个特征而言,都有相同或更好的效果。此外,本文方法节约了在模型的训练时间和检测时间。同时,本文还与其他几种模型的检测率和F-measure进行对比,结果显示本文的检测效果较好。下一步将研究如何在大数据平台上实施入侵检测,以提高在大数据时代处理海量数据的能力。

参 考 文 献

[1] Wu S, Banzhaf W. The use of computational intelligence in intrusion detection systems: A review[J]. Applied Soft Computing,2010,10(1):1-35.

[2] 欧阳广津. 基于改进的朴素贝叶斯的入侵检测方法[J]. 通信技术,2020,53(5):1273-1276.

[3] Almomani O. A feature selection model for network intrusion detection system based on PSO, GWO, FFA and GA algorithms[J]. Symmetry,2020,12(6):1046.

[4] 任学臻,张永. 基于信息增益和粗糙集的入侵检测方法[J]. 计算机应用与软件,2020,37(4):303-308.

[5] Zhao F, Zhao J, Niu X, et al. A filter feature selection algorithm based on mutual information for intrusion detection[J]. Applied Sciences,2018,8(9):1535.

[6] Natesan P, Rajalaxmi R, Gowrison G, et al. Hadoop based parallel binary bat algorithm for network intrusion detection[J]. International Journal of Parallel Programming, 2017, 45:1194-1213.

[7] Najeeb R, Dhannoon B. A feature selection approach using binary firefly algorithm for network intrusion detection system[J]. ARPN Journal of Engineering and Applied Sciences, 2018,13(6):2347-2352.

[8] Wang Z, Tang M, Deng J, et al. A new feature selection method for intrusion detection[C]//2019 IEEE International Conferences on Ubiquitous Computing & Communications and Data Science and Computational Intelligence and Smart Computing, Networking and Services,2019:298-304.

[9] Bhadra T, Mallik S, Bandyopadhyay S. Identification of multiview gene modules using mutual information-based hypograph mining[J]. IEEE Transactions on Systems, Man, and Cybernetics: Systems,2017,49(6):1119-1130.

(下转第320页)

参 考 文 献

[1] 北方网. 红色代码已给世界造成 26 亿美元损失[EB/OL]. (2001 - 09 - 03)[2021 - 01 - 27]. http://it.enorth.com.cn/system/2001/09/03/000132547.shtml? utm_source=ufqinews.

[2] 360 互联网安全中心. 2018 年勒索病毒疫情分析报告[EB/OL]. (2019 - 02 - 20)[2021 - 01 - 27]. http://zt.360.cn/1101061855.php? dtid=1101062360&did=610102590 2018.

[3] 张瑜,刘庆中,石元泉,等. 受基因理论启发的计算机病毒进化模型[J]. 电子科技大学学报,2018,47(6):888 - 894.

[4] 李延香,袁辉. 网络环境下计算机病毒及其防御技术的研究与实施[J]. 自动化技术与应用,2016,35(7):36 - 38,64.

[5] 万子龙. 勒索病毒攻击原理及检测方法研究[J]. 江西通信科技,2019(3):42 - 44.

[6] Zhu Q, Cen C. A novel computer virus propagation model under security classification[J]. Discrete Dynamics in Nature and Society,2017(83):1 - 11.

[7] Soodeh H, Azgomi M. The dynamics of an SEIRS-QV malware propagation model in heterogeneous networks[J]. Statistical Mechanics and its Applications, 2018, 512: 803 - 817.

[8] Yang L, Yang X, Zhu Q, et al. A computer virus model with graded cure rates[J]. Nonlinear Analysis: Real World Applications,2013,14(1):414 - 422.

[9] Pastor-Satorras R, Alessandro V. Epidemic spreading in scale-free networks[J]. Physical Review Letters, 2000, 86(14):3200 - 3203.

[10] Barabaási A, Albert R. Emergence of scaling in random networks[J]. Science,1999,286(5439):509 - 512.

[11] 汪小帆,李湘,陈关荣. 网络科学导论[M]. 北京:高等教育出版社,2012:3 - 22.

[12] Lazfi S, Lamzabi S, Rachadi A, et al. The impact of neighboring infection on the computer virus spread in packets on scale-free networks[J]. International Journal of Modern Physics B,2017,31(30):1750228.

[13] 曾凤琳,温罗生. 二部无标度网络上病毒传播模型和免疫策略研究[J]. 计算机工程,2014,40(8):123 - 127.

[14] Zhang C, Huang H. Optimal control strategy for a novel computer virus propagation model on scale-free networks[J]. Physica A: Statistical Mechanics and Its Applications, 2016,451:251 - 265.

[15] Yang L, Yang X, Liu J, et al. Epidemics of computer viruses: A complex-network approach[J]. Applied Mathematics and Computation,2013,219(16):8705 - 8717.

[16] 瑞星网. 2018 勒索病毒全面分析报告[EB/OL]. (2018 - 11 - 23)[2021 - 01 - 27]. http://it.rising.com.cn/fanglesuo/19459.html.

[17] Fu M, Feng J, Yang H, et al. Preferential information dynamics model for online social networks[J]. Physica A: Statistical Mechanics and Its Applications, 2018, 506: 993 - 1005.

(上接第 312 页)

[10] Rojas-Lima J, Domínguez-Pacheco A, Hernández-Aguilar C, et al. Statistical analysis of photopyroelectric signals using histogram and kernel density estimation for differentiation of maize seeds[J]. International Journal of Thermophysics,2016,37:98.

[11] Ross B. Mutual information between discrete and continuous data sets[J]. PLoS One,2014,9(2):e87357.

[12] Kraskov A, Stögbauer H, Grassberger P. Estimating mutual information[J]. Physical Review E,2004,69(6):066138.

[13] Ambusaidi M, He X, Nanda P, et al. Building an intrusion detection system using a filter-based feature selection algorithm[J]. IEEE Transactions on Computers,2016,65(10): 2986 - 2998.

[14] Yang X. Firefly algorithms for multimodal optimization[C]//International Symposium on Stochastic Algorithms, 2009:169 - 178.

[15] Selvakumar B, Muneeswaran K. Firefly algorithm based feature selection for network intrusion detection[J]. Computers & Security,2019,81:148 - 155.

[16] Mohammadi S, Mirvaziri H, Ghazizadeh-Ahsaee M. Multivariate correlation coefficient and mutual information-based feature selection in intrusion detection[J]. Information Security Journal: A Global Perspective, 2017, 26(5): 229 - 239.

[17] Al-Yaseen W. Improving intrusion detection system by developing feature selection model based on firefly algorithm and support vector machine[J]. IAENG International Journal of Computer Science,2019,46:534 - 540.

[18] Bostani H, Sheikhan M. Hybrid of binary gravitational search algorithm and mutual information for feature selection in intrusion detection systems[J]. Soft Computing, 2017, 21:2307 - 2324.

[19] Yin C, Zhu Y, Fei J, et al. A deep learning approach for intrusion detection using recurrent neural networks[J]. IEEE Access,2017,5:21954 - 21961.

[20] Aburomman A, Reaz M. A novel weighted support vector machines multiclass classifier based on differential evolution for intrusion detection systems[J]. Information Sciences, 2017,414:225 - 246.