Journal of System Simulation

Volume 33 | Issue 2 Article 11

2-20-2021

An Intrusion Detection Algorithm Based on IFOA and WELM

Jianwu Dang
School of Software and Internet of Things Engineering, Jiangxi University of Finance and Economics, Nanchang 330013, China;

Tan Ling
School of Software and Internet of Things Engineering, Jiangxi University of Finance and Economics, Nanchang 330013, China;

An Intrusion Detection Algorithm Based on IFOA and WELM

Abstract

Abstract: *An intrusion detection algorithm of WELM optimized by IFOA is proposed. The advantages of short training time and good generalization performance of WELM are used, and the weight of minority attacks is increased,* so that the recall rate of minority attacks in network attacks is greatly improved.*The FOA with adaptive adjustment of the iterative step size is used, so the input weights and bias of the hidden layer in the WELM are globally optimized* to avoid the algorithm falling into local optimal solution and realize the classification of the NSL-KDD intrusion detection data set. The experimental results show that the proposed algorithm improves the recall rate of minority attacks and the accuracy of the overall classification, and reduces the false positive rate.

Keywords

intrusion detection, unbalanced data set, weighted extreme learning machine, fruit fly optimization algorithm

Recommended Citation

Dang Jianwu, Tan Ling. An Intrusion Detection Algorithm Based on IFOA and WELM[J]. Journal of System Simulation, 2021, 33(2): 331-338.

第 33 卷第 2 期 系统仿真学报© Vol. 33 No. 2
2021 年 2 月 Journal of System Simulation Feb. 2021

改进果蝇算法优化加权极限学习机的入侵检测

党建武，谭凌
(江西财经大学 软件与物联网工程学院，江西 南昌 330013)

摘要：提出一种改进的果蝇算法优化加权极限学习机入侵检测算法，利用加权极限学习机训练时间短、泛化性能好等优点，对NSL-KDD入侵检测数据集中的不均衡现象，增加少数类攻击的权重，使对网络攻击中稀有攻击的检测率比传统机器学习方法有大幅提高；用迭代步长自适应调整的果蝇优化算法，对加权极限学习机中的隐含层输入权值和偏置进行全局寻优，以避免算法陷入局部最优解，实现了对NSL-KDD入侵检测数据集的分类。实验表明：本算法对稀有攻击的检测率和分类准确率均有提高，误报率有所降低。

关键词：入侵检测；不均衡数据集；加权极限学习机；果蝇优化算法
中图分类号：TP391.9 文献标志码：A 文章编号：1004-731X (2021) 02-0331-08
DOI: 10.16182/j.issn1004731x.joss.19-0361

An Intrusion Detection Algorithm Based on IFOA and WELM

Dang Jianwu, *Tan Ling*
(School of Software and Internet of Things Engineering, Jiangxi University of Finance and Economics, Nanchang 330013, China)

Abstract: *An intrusion detection algorithm of WELM optimized by IFOA is proposed. The advantages of short training time and good generalization performance of WELM are used, and the weight of minority attacks is increased,* so that the recall rate of minority attacks in network attacks is greatly improved.*The FOA with adaptive adjustment of the iterative step size is used, so the input weights and bias of the hidden layer in the WELM are globally optimized* to avoid the algorithm falling into local optimal solution and realize the classification of the NSL-KDD intrusion detection data set. The experimental results show that the proposed algorithm improves the recall rate of minority attacks and the accuracy of the overall classification, and reduces the false positive rate.

Keywords: intrusion detection; unbalanced data set; weighted extreme learning machine; fruit fly optimization algorithm

引言

机器学习在入侵检测中的研究受到了网络安全领域研究人员的高度关注，并在其理论发展、关键技术研究及应用等方面取得了一定的成果[1-2]。应用于入侵检测中的机器学习算法主要有：贝叶斯分类算法[3]、支持向量机[4-5]以及 BP 神经网络算法[6-8]等。文献[9]提出了一种有效的协同入侵检测网络，在该网络中，每个入侵检测系统使用贝叶斯学习来评估相邻近系统的入侵检测率和误报率，并使用贝叶斯决策模型对结果进行聚类，以最小化错误的入侵决策和维护的成本。文献[10]提出了一种并行的入侵检测模型，该模型先用 ReliefF 算法进行数据的去燥处理，然后采用改进的乌鸦搜索算法并行的进行数据的降维操作和对核极限学习机参数的优化，最后将训练好的核极限学习机用于对

收稿日期：2019-07-29 修回日期：2019-11-28
第一作者：党建武(1964-)，男，博士，教授，研究方向为智能信息处理、信息安全、金融时间序列分析。E-mail: dangjianwu006@163.com
通讯作者：谭凌(1995-)，女，硕士生，研究方向为信息安全。E-mail: tanling@jjccb.com

测试集的分类。结果表明，该模型对入侵检测的分类精度和速度都有很大的提升。文献[11]中，一种新型单隐层前馈型神经网络模型极限学习机(Extreme Learning Machine，ELM)被黄广斌等提出，该模型相比于传统的 BP 神经网络无需反向迭代来调整参数，具有学习时间相对较短、泛化性能较好等优点。文献[12]就 BP 神经网络算法在入侵检测中出现的收敛速度慢以及检测率低的这些问题，提出了一种基于主成分分析(PCA)和极限学习机(ELM)的入侵检测算法。与传统的机器学习算法相比，该算法在入侵检测率、正确率、漏报率以及误报率等评价指标上均有所改善。

ELM 自从提出以来受到较多关注，其在入侵检测上也体现出了较好的性能[13-14]，所存在的问题是当数据种类分布不均衡时，该方法对分类的效果会出现大幅下降。在现有用于入侵检测上的机器学习算法中，对网络流量中的数据不均衡问题较少考虑。本文提出一种改进的果蝇算法优化加权极限学习机的入侵检测算法，算法利用加权极限学习机[15]作为前馈型神经网络时具有训练时间短、泛化性能好等优点，对于 NSL-KDD 入侵检测数据集中的不均衡现象，增加少数类攻击的权重，使得对网络攻击中少数类攻击的检测率相较于传统的机器学习有大幅度的提高；利用改进后果蝇优化算法强大的全局寻优能力，对加权极限学习机中的隐含层输入权值和偏置进行全局寻优，以避免算法陷入局部最优解，实现了对 NSL-KDD 入侵检测数据集的分类。

1 加权极限学习机

本文以 ELM 为基础，建立一个加权极限学习机分类模型。ELM 模型的网络结构如图 1 所示。

图 1 中，假设给定 N 个训练样本 $\{x_i,t_i\}_{i=1}^{N}$，其中，$x_i=[x_{i1},x_{i2},\cdots,x_{in}]^{\mathrm{T}}\in R^n$，$t_i=[t_{i1},t_{i2},\cdots,t_{im}]^{\mathrm{T}}\in R^m$，$n$ 为样本的特征数，m 为样本的类别数。一个含有 L 个隐含层节点的前馈神经网络输出模型可以表示如下：

$$\sum_{h=1}^{L}\beta_h G(a_h,b_h,x)=o_i,\ i=1,2,\cdots,N$$

式中：β_h 为第 h 个隐含层神经元的输出权值；G 为隐含层神经元的激活函数；a_h，b_h 分别为第 h 个隐含层神经元的输入权值和偏置；x 为输入样本，o_i 为第 i 个训练样本的实际输出值；t_i 为第 i 个训练样本的期望输出。

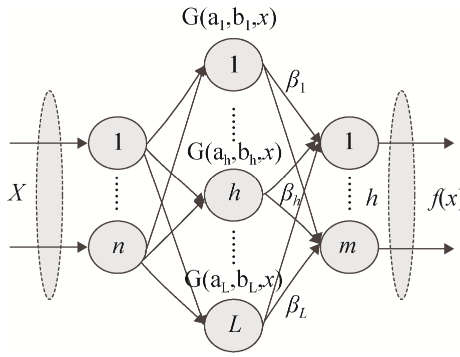

图 1 单隐层前馈神经网络基本结构图
Fig. 1 Basic structure diagram of single hidden layer feedforward neural network

根据文献[15]，对于一个数量为 N 的训练样本，$\{x_i,t_i\}_{i=1}^{N}$，$x_i\in R^n$，存在一个(a_h，b_h)和 β_h 有 $\sum_{i=1}^{L}\|o_i-t_i\|=0$，使得该单隐含层前馈神经网络模型(single-hidden layer feedforward network, SLFN)能够以零误差去逼近训练集 $\{x_i,t_i\}_{i=1}^{N}$，$x_i\in R^n$，即

$$\sum_{h=1}^{L}\beta_h G(a_h,b_h,x_i)=t_i,i=1,2,\cdots,N \tag{1}$$

式(1)可进一步化简为：

$$\boldsymbol{H\beta}=\boldsymbol{T}$$

式中：$\boldsymbol{H}$ 为隐含层输出矩阵；$\boldsymbol{\beta}$ 为隐含层输出权值矩阵；$\boldsymbol{T}$ 为训练样本对应的期望输出矩阵。

ELM 在训练过程中，初始化网络参数时，随机生成隐含层的输入权值 a_h 和隐含层偏置 b_h，并且在整个训练和测试的过程中维持不变。由于输入的训练样本、隐含层的输入权值与偏置、期望输出均已知，则整个训练过程就是为了求出 ELM 模型中的隐含层输出权值矩阵 $\boldsymbol{\beta}$，从而得到完整的分类模型。

https://dc-china-simulation.researchcommons.org/journal/vol33/iss2/11
DOI: 10.16182/j.issn1004731x.joss.19-0361

由隐含层输出矩阵 $\boldsymbol{H}$ 的 Moore-Penrose 广义逆矩阵 $\boldsymbol{H}^{+}$可解得

$$\hat{\boldsymbol{\beta}}=\boldsymbol{H}^{+}\boldsymbol{T} \tag{2}$$

式中：$\boldsymbol{H}^{+}$的计算方式有多种，在 ELM 中，通常采用正交投影法(KKT)对 $\boldsymbol{H}^{+}$求解。当 $\boldsymbol{H}^{\mathrm{T}}\boldsymbol{H}$ 为非奇异矩阵时，$\boldsymbol{H}^{+}=(\boldsymbol{H}^{\mathrm{T}}\boldsymbol{H})^{-1}\boldsymbol{H}^{\mathrm{T}}$；当 $\boldsymbol{H}\boldsymbol{H}^{\mathrm{T}}$ 为非奇异矩阵时，$\boldsymbol{H}^{+}=\boldsymbol{H}^{\mathrm{T}}(\boldsymbol{H}\boldsymbol{H}^{\mathrm{T}})^{-1}$。

为求解式(2)，在 $\boldsymbol{H}^{\mathrm{T}}\boldsymbol{H}$ 或者 $\boldsymbol{H}\boldsymbol{H}^{\mathrm{T}}$ 的对角线上添加上一个足够小的正则项 $1/C$，使得分类模型有了更好的稳定性和泛化性能。隐含层的输出权重可以表示为

$$\hat{\boldsymbol{\beta}}=\begin{cases}\boldsymbol{H}^{\mathrm{T}}(I/C+\boldsymbol{H}\boldsymbol{H}^{\mathrm{T}})^{-1}\boldsymbol{T}, N<L\\(I/C+\boldsymbol{H}^{\mathrm{T}}\boldsymbol{H})^{-1}\boldsymbol{H}^{\mathrm{T}}\boldsymbol{T}, N\geqslant L\end{cases}$$

ELM 的输出函数可以表示为

$$f(x)=h(x)\hat{\boldsymbol{\beta}}=\begin{cases}h(x)\boldsymbol{H}^{\mathrm{T}}(I/C+\boldsymbol{H}\boldsymbol{H}^{\mathrm{T}})^{-1}\boldsymbol{T}, N<L\\h(x)(I/C+\boldsymbol{H}^{\mathrm{T}}\boldsymbol{H})^{-1}\boldsymbol{H}^{\mathrm{T}}\boldsymbol{T}, N\geqslant L\end{cases} \tag{3}$$

在分类问题中，并非所有的分类样本数据都是均衡分布的，Zong 等[15]为了解决不均衡样本的分类问题，在 ELM 的基础上提出了加权极限学习机算法(Weighted Extreme Learning Machine, WELM)。文献[16]提出根据加权方案赋予每个样本权值：

加权方案一 $\boldsymbol{W}_1$：自动加权方案：

$$\boldsymbol{W}_1=\frac{1}{\mathrm{Count}(t_i)}$$

式中：$\mathrm{Count}(t_i)$ 为训练样本中类别为 t_i 的样本数。

加权方案二 $\boldsymbol{W}_2$：将少数类和多数类的比例向 0.618:1(黄金分割比)的方向推进，这个方案实际上是以牺牲多数类的分类精度，来换取少数类的分类精度。

$$\boldsymbol{W}_2=\begin{cases}\dfrac{0.618}{\mathrm{Count}(t_i)}，t_i\text{属于多数类}\\\dfrac{1}{\mathrm{Count}(t_i)}，t_i\text{属于少数类}\end{cases}$$

WELM 隐含层的输出权重可以表示为

$$\hat{\boldsymbol{\beta}}=\boldsymbol{H}^{+}\boldsymbol{T}=\begin{cases}\boldsymbol{H}^{\mathrm{T}}(I/C+\boldsymbol{W}\boldsymbol{H}\boldsymbol{H}^{\mathrm{T}})^{-1}\boldsymbol{W}\boldsymbol{T}, N<L\\(I/C+\boldsymbol{H}^{\mathrm{T}}\boldsymbol{W}\boldsymbol{H})^{-1}\boldsymbol{H}^{\mathrm{T}}\boldsymbol{W}\boldsymbol{T}, N\geqslant L\end{cases}$$

式中：加权矩阵为 $N\times N$ 的对角矩阵；N 个主对角元素对应着 N 个样本，将不同的权值赋予不同的样本类别，其中同一类别的加权权值相同。

在隐含层特征映射 $h(x)$未知的情况下，将核矩阵定义为

$$\boldsymbol{\Omega}_{\mathrm{ELM}}=\boldsymbol{H}\boldsymbol{H}^{\mathrm{T}}:\boldsymbol{\Omega}_{\mathrm{ELM}i,j}=h(x_i)h(x_j)=K(x_i,x_j) \tag{4}$$

式中：$\boldsymbol{\Omega}_{\mathrm{ELM}}$ 为核矩阵，核函数 K 需满足 Mercer 条件，比较常见的核函数有 Gaussian 核函数、径向基核函数、多项式核函数以及线性核函数。由式(4)可将输出函数表达式(3)表示为

$$f(x)=h(x)\hat{\boldsymbol{\beta}}=h(x)\boldsymbol{H}^{\mathrm{T}}(I/C+\boldsymbol{W}\boldsymbol{H}\boldsymbol{H}^{\mathrm{T}})^{-1}\boldsymbol{W}\boldsymbol{T}=\begin{bmatrix}K(x,x_1)\\\vdots\\K(x,x_N)\end{bmatrix}(I/C+\boldsymbol{W}\boldsymbol{\Omega}_{\mathrm{ELM}})^{-1}\boldsymbol{W}\boldsymbol{T} \tag{5}$$

因此，基于加权极限学习机的分类模型的训练流程为：

(1) 随机设置隐含层的输入权值 a_h 和偏置 b_h，其中 h=1,2, ... ,L；

(2) 根据加权方案为每个样本赋予权值，计算出加权矩阵 $\boldsymbol{W}$；

(3) 根据选定的核函数，计算出核矩阵 $\boldsymbol{\Omega}_{\mathrm{ELM}}$；

(4) 利用式(5)计算输出。

2 基于改进果蝇优化加权极限学习机算法

2.1 改进的果蝇优化算法

受果蝇觅食行为的启发，潘文超提出了果蝇优化算法(Fruit Fly Optimization Algorithm, FOA)[16]，其基本思想是利用果蝇优越的视觉和嗅觉上的感知能力来确定食物的位置。其基本寻优过程可分为以下几个步骤[17-18]：

step 1：参数初始化，设置种群规模 N、最大迭代次数 max 和果蝇群体位置 X_axis, Y_axis，给出每个果蝇个体随机的方向和距离，然后果蝇个体开始利用嗅觉搜索食物：

$$\begin{aligned}X_i&=X\_axis+Rand()\\Y_i&=Y\_axis+Rand()\end{aligned} \tag{6}$$

式中：*Rand*()为果蝇的飞行范围，即迭代步长。

step 2：计算每个果蝇个体与坐标原点之间的距离 $Dist_i$，再计算每个果蝇个体的味道浓度判定值 S_i：

$$Dist_i = \sqrt{X_i^2 + Y_i^2}$$

$$S_i = 1/Dist_i \text{。}$$

step 3：将 step 2 中的味道浓度判定值 S_i 代入味道浓度判定函数(Fitness *Function*)，求出每个果蝇个体位置的味道浓度 $Smell_i$，并找出该果蝇群体中味道浓度最佳的果蝇(求极大值)：

$$Smell_i = Function(S_i) \tag{7}$$

$$[\mathrm{best}Smell\ \mathrm{best}Index] = \max(Smell_i) \text{。}$$

step 4：记录下味道浓度最佳果蝇的味道浓度值以及其位置坐标，此时果蝇群体发挥其视觉优势逐渐飞向这个位置：

$$\begin{aligned} &Smellbest = \mathrm{best}Smell \\ &X\_axis = X(\mathrm{best}Index) \\ &Y\_axis = Y(\mathrm{best}Index) \text{。} \end{aligned} \tag{8}$$

step 5：进入迭代优化的阶段，重复上述 step 2~3，并判断味道浓度值是否大于前一迭代的味道浓度。若否，则在最大迭代次数内继续重复上述 step 2~3；若是，则去执行 step 4。

在实际的应用过程当中，果蝇优化算法的搜索距离通常是固定值，缺少自适应性，从而导致 FOA 的全局搜索能力下降，并且容易陷入局部搜索[19]。为了解决果蝇优化算法所存在的不足，本文对果蝇优化算法的搜索距离进行了相应的改进。在果蝇的前期搜索过程中先给予一个最大范围的搜索半径，便于迅速地在全局范围内寻得较优解，然后，随着迭代次数的增加再逐渐缩小搜索半径，从而在小范围内进行较为精准的搜索，最后确定最优解。因此，本文对式(6)进行改进，如式(9)所示：

$$\begin{aligned} X_i &= X\_axis + \alpha^k \times Rand() \\ Y_i &= Y\_axis + \alpha^k \times Rand() \end{aligned} \tag{9}$$

式中：α 为步长控制因子[20]；k 为迭代次数。随着迭代次数的增加，果蝇的搜索范围将呈指数形式的下降。

2.2 基于 IFOA-WELM 的入侵检测分类算法

本文利用 WELM 作为前馈型神经网络时具有训练时间短、泛化性能好等优点及 IFOA 的全局寻优能力，使用 IFOA 算法对 WELM 网络中随机确定的隐含层输入权值和偏置进行优化处理，既解决了网络入侵检测中的数据不均衡问题，使得对网络攻击中少数类攻击的召回率有大幅度的提高，又避免了 WELM 陷入局部最优解。本文算法流程见图 2。

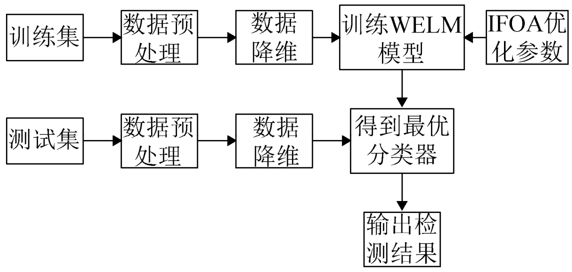

图 2　本文算法流程图
Fig.2　Algorithm flow chart

具体的算法流程如下：

(1) 从 NSL-KDD 数据集中选取训练集和测试集，并对数据集中的数据进行预处理；

(2) 分析数据集中各个特征之间的相关性，选取相关性较大的部分特征对数据集进行降维处理；

(3) 初始化 IFOA 和 WELM 的网络参数：随机确定果蝇群体的初始位置[*X_axis*, *Y_axis*]，设置种群规模 N，步长控制因子 α 和最大迭代次数 max，WELM 的输入神经元、隐含层神经元以及输出神经元个数，确定加权方案，将 WELM 隐含层的输入权值和偏置初始化；

(4) 将训练集输入 WELM，根据式(7)计算出个体果蝇的味道浓度；

(5) 寻找果蝇群体中味道浓度最佳的果蝇个体，并根据式(8)记录其位置；

(6) 根据式(9)更新果蝇的位置和飞行方向，进入迭代寻优阶段；

(7) 若迭代次数大于 max，则保存下最优味道浓度的果蝇个体的位置，即全局最优的隐含层输入权值和偏置；否则，迭代次数加一，返回步骤(2)；

(8) 将最优的隐含层输入权值和偏置代入 WELM 中，对测试集进行实验。

3 算法仿真与结果分析

3.1 数据集的选择及预处理

本文选用 NSL-KDD 数据集[21]作为实验数据集，选择 NSL-KDD 数据包中的 KDDTrain+和 KDDTest+分别作为实验的训练集和测试集。每个数据集有 42 维数据，其中，前 41 维为数据集特征，第 42 维为数据集标签位。标签位包括正常数据 Normal 和 39 种攻击类型，其中这 39 种攻击的类型分别属于 DOS、U2R、R2L 以及 Probe 这四大类攻击。其中，训练集包括了 21 种类型的入侵攻击，测试集中则出现了 18 种训练集中没有入侵攻击。这些只出现在测试集的入侵攻击，能够用来评价本文的入侵检测算法对未知攻击的检测能力。表 1 为训练集和测试集中各个标签类的分布情况。

表 1 训练集和测试集中各个标签类的分布情况
Tab. 1 Distribution of each tag class in training set and test set

标签类	Normal	Dos	U2R	R2L	probe	总
训练集	67 343	45 927	52	995	11 656	125 973
测试集	9 711	7 458	200	2 754	2 421	22 544

训练模型前，需要对数据集中的数据进行预处理：

(1) 将特征中的字符型转化为数值型

NSL-KDD 数据集的 41 维特征中，第 2 维特征协议类型(Protocol Type)、第 3 维特征网络服务(Service)以及第 4 维特征连接状态(Flag)均为字符型特征，需转化为数值型。将第 2 维特征中的 TCP 记为 1、UDP 记为 2、TCMP 记为 3，第 3 维特征中的 67 种 Service 类型分别按照其名称首字母顺序记为 1~67，第 4 维特征中的 11 种 Flag 状态分别记为 1~11。第 42 维标签位中，共有 5 类标签：Normal, DOS, Probe, U2R 以及 R2L，分别将其记为 0~4。

(2) 数据标准化和归一化

将上一步骤中数值化后的数据集根据式(10)和式(11)进行相应的处理，统一不同特征值之间的度量单位，以减少因度量单位的差异而给检测结果带来的影响。

标准化公式：

$$x_1=\left(x-\overline{x}\right)/\sigma \tag{10}$$

式中：x 为特征值；$\overline{x}$ 为该特征值的平均值；σ 为该特征值的标准差；x_1 为每个数据样本该维特征标准化后的结果。

归一化公式为：

$$x_2=\left(x_1-x_{1\min}\right)/\left(x_{1\max}-x_{1\min}\right) \tag{11}$$

式中：$x_{1\min}$ 为该维特征的所有样本经式(10)处理后出现的最小值；$x_{1\max}$ 为该维特征的所有样本经式(10)处理后出现的最大值；x_2 为每个数据样本该维特征归一化后的结果。

3.2 实验评测指标

本文采用准确率、误报率和召回率这三项性能评价指标对本文算法的优劣性进行评价。

(1) 准确率 Accuracy：$\mathrm{ACY}=\dfrac{\mathrm{TP}+\mathrm{TN}}{\mathrm{TP}+\mathrm{FP}+\mathrm{TN}+\mathrm{FN}}$

(2) 误报率 False Positive Rate：$\mathrm{FPR}=\dfrac{\mathrm{FP}}{\mathrm{FP}+\mathrm{TN}}$

(3) 召回率 Recall：$\mathrm{RC}=\dfrac{\mathrm{TP}}{\mathrm{TP}+\mathrm{FN}}$

式中：TP(True Positive)真正例：实际为入侵攻击，预测结果也为入侵攻击的数据量(被正确识别为正常数据的样本数)；TN(Ture Negative)真负例：实际为正常数据，预测结果也为正常数据的数据量(被正确识别为攻击数据的样本数)；FP(False Positive)假正例：实际为正常数据，预测结果为入侵攻击的数据量(被错误识别为正常数据的样本数)；FN(False Negative)假负例：实际为入侵攻击，预测结果为正常数据的数据量(被错误识别为攻击数据的样本数)。

3.3 数据降维处理

将预处理后的 41 维数据特征采用皮尔逊相关系数的方法对其进行相关性分析，得到各个特征的

权重排序图，其中排列顺序按照 NSL-KDD 数据集中特征的顺序依次排列，如图 3 所示。

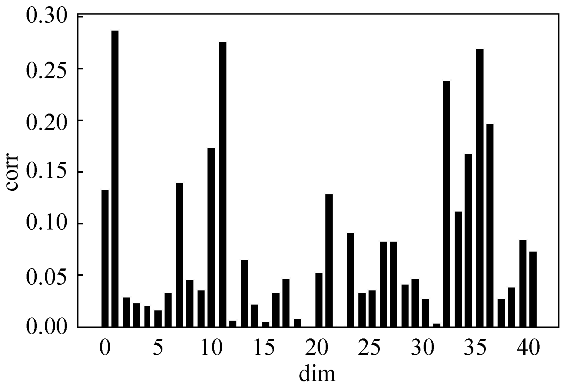

图 3 特征权重排序图
Fig. 3 Sorting graph of feature weights

实验中，取相关系数(corr)>0.05 的特征，作为降维后的训练集和测试集。因此，降维后的训练集和测试集中包含 18 维特征。

3.4 仿真与结果分析

本文仿真环境为 inter i5-3470 处理器 3.2GHz，4GB 内存，用 Python 编制仿真程序来实现。

(1) WELM 与其他机器学习算法对比实验

设置 WELM 输入层节点数为 18，隐含层节点数为 250，输出层节点数为 5，加权方案选择方案 2，核函数选择 Sigmoid 核函数。将本文算法检测结果与 K 最近邻算法(KNN)、随机森林(Random Forest, RF)、BP 神经网络以及极限学习机(ELM)算法检测的结果进行一个对比实验。对比实验结果见表 2。

由表 2 可以看出，加权极限学习机算法在 Normal、Dos 以及 Probe 这 3 类的检测效果上与其他机器学习算法相差无几，而在 U2R 和 R2L 这两类的检测效果上，WELM 的召回率远高于其他机器学习算法。但相比之下，这两类的召回率还是相对较低，这是因为 U2R 和 R2L 这两大攻击在实际生活中出现较少，所以在训练集中的样本量太少，数据集出现不均衡现象。

(2) WELM 与 IFOA-WELM 算法的对比实验

将分别采用 IFOA 和 FOA 优化了的 WELM 算法与没有优化的 WELM 算法进行对比试验，设置果蝇种群大小为 50，迭代次数为 300 次，果蝇飞行的方向和距离设置为[–20, 20]，步长控制因子 α 为 0.95。WELM 的输入层节点个数为 18，隐含层节点个数为 250，输出层节点数为 5，加权方案选择方案 2，核函数选择 Sigmoid 核函数。实验结果见表 3。

表 2 不同算法的检测效果对比表
Tab. 2 Comparison table of detection effects of different algorithms /%

算法	召回率					测试集分类准确率	误报率
	Nomal	Dos	U2R	R2L	Probe		
KNN	96	75	12	5	56	79	30.5
RF	96	77	46	15	68	82	30.1
BP	94	71	0	34	67	80	28.2
ELM	96	77	0	37	56	81	11.3
WELM	93	75	65	67	70	83	6.6

表 3 WELM 与 FOA-WELM 对比实验结果
Tab. 3 Comparison of experimental results between WELM and IFOA-WELM

算法	召回率/%					测试集分类准确率/%	误报率/%	训练时间/s
	Nomal	Dos	U2R	R2L	Probe			
WELM	93	75	65	67	70	83	6.6	2.52
FOA-WELM	93	80	65	70	71	85	4.1	2.10
IFOA-WELM	93	80	71	70	74	85	3.8	2.13

DOI: 10.16182/j.issn1004731x.joss.19-0361

从表 3 可以看出，采用 FOA 或 IFOA 优化后的 WELM 算法经过了全局寻优，使得四大类攻击的召回率均有所提升。而相较于 FOA-WELM，IFOA-WELM 得到的分类效果更好，尤其是 U2R 攻击的召回率相较于 WELM 提高了 6%，测试集分类准确率提高了 2%，误报率降低了 2.8%。在实验环境相同的情况下，设置果蝇种群大小为 50，迭代次数为 300 次，果蝇飞行的方向和距离设置为[–20, 20]，步长控制因子 α 为 0.95。FOA-WELM 的训练时长为 2.10 s，IFOA-WELM 算法的训练时长为 2.13 s，这是由于新增了优化算法的自适应性，提高了时间复杂度，所以训练时间有所增加。

4 结论

本文针对入侵检测中的数据不均衡问题，利用 WELM 算法增加少数类的检测权重，同时，利用 FOA 算法对其进行寻优，在一定程度上提高了入侵检测中对少数类的召回率，仿真结果表明，与传统的机器学习算法相比，WELM 算法对两大少数类攻击的召回率均有一定程度的提高，误报率随之降低，因此，WELM 更适用于入侵检测的研究；利用 IFOA 算法优化后的 WELM 算法，在召回率和误报率方面均有进一步的提高，训练时间得到减少，提高了入侵检测的实时性。

参考文献：

[1] 康松林, 刘乐, 刘楚楚, 等. 多层极限学习机在入侵检测中的应用[J]. 计算机应用, 2015, 35(9): 2513-2518.
Kang Songlin, Liu Le, Liu Chuchu, et al. Intrusion Detection Based on Multiple Layer Extreme Learning Machine[J]. Journal of Computer Application, 2015, 35(9): 2513-2518.

[2] Akashdeep, Manzoor I, Kumar N. A Feature Reduced Intrusion Detection System Using ANN Classifier[J]. Expert Systems With Applications (S0957-4174), 2017, 88(12): 249-257.

[3] 姚潍, 王娟, 张胜利. 基于决策树与朴素贝叶斯分类的入侵检测模型[J]. 计算机应用, 2015, 35(10): 2883-2885.
Yao Wei, Wang Juan, Zhang Shengli. Intrusion Detection Model Based on Decision Tree and Naive-Bayes Classification[J]. Journal of Computer Application, 2015, 35(10): 2883-2885.

[4] Chitrakar R, Huang C H. Selection of Candidate Support Vectors in Incremental SVM for Network Intrusion Detection[J]. Computers and Security (S0167-4048), 2014, 45(3): 231-241.

[5] Gu J, Wang L H, Wang H W, et al. A Novel Approach to Intrusion Detection Using SVM Ensemble with Feature Augmentation[J]. Computers and Security (S0167-4048), 2019, 86(9): 53-62.

[6] 沈夏炯, 王龙, 韩道军. 人工蜂群优化的 BP 神经网络在入侵检测中的应用[J]. 计算机工程, 2016, 42(2): 190-194.
Shen Xiajiong, Wang Long, Han Daojun. Application of BP Neural Network Optimized by Artificial Bee Colony in Intrusion Detection[J]. Journal of Computer Engineering, 2016, 42(2): 190-194.

[7] 丁红卫, 万良, 邓烜堃. 改进的 HS 算法优化 BP 神经网络的入侵检测研究[J]. 计算机工程与科学, 2019, 41(1): 65-72.
Ding Hongwei, Wan Liang, Deng Xuankun. Optimizing Intrusion Detection of A Modified Harmony Search Algorithm[J]. Journal of Computer Engineering and Science, 2019, 41(1): 65-72.

[8] Zhu Y Z. Intrusion Detection Method Based on Improved BP Neural Network Research[J]. International Journal of Security and Its Applications (S1738-9976), 2016, 10(5): 193-202.

[9] Fung C J, Zhang J, Boutaba R. Effective Acquaintance Management Based on Bayesian Learning for Distributed Intrusion Detection Networks[J]. IEEE Transactions on Network and Service Management (S1932-4537), 2012, 9(3): 320-332.

[10] 马超. 基于 ReliefF 和改进乌鸦搜索优化的并行入侵检测方法[J]. 计算机应用研究, 2019, 36(10): 3063-3068.
Ma Chao. Network Intrusion Based on ReliefF and Improved Crow Search Optimization Parallel Method[J]. Journal of Application Research of Computers, 2019, 36(10): 3063-3068.

[11] Huang G B, Zhu Q Y, Siew C K. Extreme learning machine: Theory and application[J]. Neurocomputing (S0925-2312), 2006, 70(1/3): 489-501.

[12] 黄思慧, 陈万忠, 李晶. 基于 PCA 和 ELM 的网络入侵检测技术[J]. 吉林大学学报(信息科学版)，2017, 35(5): 576-583.
Huang Sihui, Chen Wanzhong, Li Jing. Network

Intrusion Detection Based on Extreme Learning Machine and Principal Component Analysis[J]. Journal of Jilin University (Information Science Edition), 2017, 35(5): 576-583.

[13] 王琳琳, 刘敬浩, 付晓梅. 基于极限学习机与改进 K-means 算法的入侵检测方法[J]. 计算机工程与科学, 2018, 40(8): 1398-1404.
Wang Linlin, Liu Jinghao, Fu Xiaomei. An Intrusion Detection Method Based on Extreme Learning Machine and Modified K-means[J]. Journal of Computer Engineering and Science, 2018, 40(8): 1398-1404.

[14] Al-Yaseen W L, Othman Z A, Nazri M Z A. Multi-level Hybrid Support Vector Machine and Extreme Learning Machine Based on Modified K-means for Intrusion Detection System[J]. Expert Systems with Applications (S0957-4174), 2017, 67(1): 296-303.

[15] Zong W W, Huang G B, Chen Y Q. Weighted Extreme Learning Machine for Imbalance Learning[J]. Neurocomputing (S0925-2312), 2013, 101(3): 229-242.

[16] Pan W T. A New Fruit Fly Optimization Algorithm: Taking the Financial Distress Model as An Example[J]. Knowledge-based System (S0950-7051), 2012, 26(2): 69-74.

[17] Lv S X, Zeng Y R, Wang L. An Effctive Fruit Fly Optimization Algotithm with Hybrid Information Exchange and Its Applications[J]. International Journal of Machine Learning and Cybernetics (S1868-8071), 2018, 9(10): 1623-1648.

[18] Hu R, Wen S, Zeng Z. A Short-term Power Load Forecasting Model Based on the Generalized Regression Neural Network With Decreasing Step Fruit Fly Optimization Algorithm[J]. Neuroconputing (S0925-2312), 2017, 221(1): 24-31.

[19] 李少波, 赵辉, 赵成龙, 等. 果蝇优化算法研究综述[J]. 科学技术与工程, 2018, 18(1): 163-171.
Li Shaobo, Zhao Hui, Zhao Chenglong, et al. Review of Fruit Fly Optimization Algorithms[J]. Journal of Science Technology and Engineering, 2018, 18(1): 163-171.

[20] Shan D, Cao G H, Dong H J. LGMS-FOA: An Improved Fruit Fly Optimization Algorithm for Solving Optimization Problems[J]. Mathematical Problems in Engineering (S1563-5147), 2013(7): 1256-1271.

[21] Dhanabal L, Shantharajah S P. A Study on NSL-KDD Dataset for Intrusion Detection System Based on Classification Algorithms[J]. International Journal of Advanced Research in Computer and Communication Engineering (S2319-5940), 2015, 4(6): 446-452.