

引用格式:赵宏,常兆斌,王伟杰. 基于深度自编码和决策树的恶意域名检测[J]. 微电子学与计算机, 2020, 37(5): 13-17.
[ZHAO H, CHANG Z B, WANG W J. Malicious domain name detection based on deep auto-encoder and decision tree[J]. Microelectronics & Computer, 2020, 37(5): 13-17.]

基于深度自编码和决策树的恶意域名检测

赵 宏,常兆斌,王伟杰

(兰州理工大学 计算机与通信学院,甘肃 兰州 730050)

摘 要: 针对目前恶意域名检测方法特征提取过程复杂和检测准确率不高的问题,提出一种基于深度自编码和决策树(Deep Auto Encoder and Decision Tree, DAE-DT)的恶意域名检测算法. 该算法首先将每一域名按照域名词法组成与结构等属性进行特征映射,并进行正则化处理;然后将正则化处理后的无标签域名数据随机置 0 作为模型的输入,域名字符统计特征作为输出,构造深度自编码网络模型. 并通过计算模型输出值与未处理数据之间的重构误差,实现各层参数与权值的优化,以增强模型的鲁棒性;最后依据提取的域名字符统计特征构造恶意域名判定的决策树. 通过在 Alexa 和 Malware domain list 等标准数据集上进行测试. 实验结果表明,该模型的检测准确率、精确率、假阴性率和假阳性率值分别为 95.21%、94.17%、2.41% 和 3.63%.

关键词: 恶意域名检测;深度自编码;决策树;域名统计特征;重构误差

中图分类号: TP393

文献标识码: A

文章编号: 1000—7180(2020)05—0013—05

Malicious domain name detection based on deep auto-encoder and decision tree

ZHAO Hong, CHANG Zhao-bin, WANG Wei-jie

(School of Computer and Communication, Lanzhou University of Technology, Lanzhou 730050, China)

Abstract: Aiming at the problem that the existing malicious domain name detection methods are not effective enough in performance of accuracy rate and the process of feature extraction, a malicious domain name detection algorithm based on deep auto-encoder and decision tree (DAE-DT) is proposed. According to lexical composition and structure of domain name, each domain name is firstly mapped into the feature space and it is normalized. Then the normalized unlabeled domain names are randomly set to 0 as the input of the model, and the statistical features of domain name are used to as the output to construct the deep auto-encoder network model, and the reconstruction error of the unprocessed data and output data is computed to achieve the purpose of optimizing the parameters and weights so that the model is more robust. Finally, a decision tree for malicious domain name detection is constructed based on the statistical features of domain name. In the experiments on Alexa and malware domain list, the proposed detection algorithm yield an accuracy rate of 95.21%, a precision rate of 94.17%, a false negative rate of 2.41%, and a false positive rate of 3.63%.

Key words: malicious domain name detection; deep auto-encoder network; decision tree; statistical feature of domain name; reconstruction error

收稿日期: 2019—09—20; 修回日期: 2019—10—16

基金项目: 国家自然科学基金项目(51668043, 61262016); 赛尔网络下一代互联网技术创新项目(NGII20160311, NGII20160112); 兰州理工大学学生科技创新基金(KC2019ZR016)

1 引言

互联网技术的快速发展给人们的工作与生活带来了许多便利,同时利用网络通信服务系统进行违法犯罪的事件不断增多^[1].

域名系统(Domain Name System, DNS)作为互联网中实现网络域名与IP地址相互转换的一种基础服务,得到了广泛应用,如何高效的检测出恶意域名已成为网络安全领域的一个重要研究课题^[2].

近年来,机器学习与深度学习算法广泛应用于网络空间安全领域^[3-6],本文提出了一种基于深度自编码和决策树的恶意域名检测算法,该算法首先将每一域名按照域名词法组成与结构等属性进行特征映射,并进行正则化处理;然后利用随机映射函数将正则化处理后的无标签域名数据作为输入,域名字符统计特征作为输出,构造深度自编码网络模型,并通过计算模型输出值与未处理数据之间的重构误差,实现各层参数与权值的优化,以增强模型的鲁棒性;最后依据提取的域名字符统计特征构造恶意域名判定的决策树.

2 算法设计与实现

基于深度自编码网络和决策树的恶意域名检测算法包括数据标准化、深度自编码网络构造、决策树训练与预测等三部分.其中数据标准化是将域名按照字符属性值转化到 $[0, 1]$ 区间,降低由于特征数据间差异性较大等原因带来的检测误差;深度自编码网络模型构造主要包括单层自编码器叠加、预训练和参数微调等三部分;决策树训练与预测是将深度自编码网络提取的域名字符统计特征作为构造决策树的先决条件,实现恶意域名的快速判定.

2.1 自动编码器

自编码网络(Auto Encoder, AE)^[7]是一种基于反向传播思想设计的无监督学习算法,该网络通过将原始输入数据压缩成潜在空间特征,然后根据该类特征重构出输入,并根据原始输入数据与目标输出之间的重构误差来优化网络各层参数,实现输入数据中关键特征的提取.自编码网络实现特征提取主要包括编码和解码等两个阶段.

编码阶段是将原始高维数据降维,压缩为潜在低维特征;解码阶段根据降维后的数据特征重构原始输入.编码和解码过程如公式(1)和(2)所示.

$$h_1 = f(x) = \sigma_1(xW_1 + b_1) \quad (1)$$

$$o_2 = g(h_1) = \sigma_2(hW_2 + b_2) \quad (2)$$

式中, W_1, b_1 为编码权重和偏置; W_2, b_2 为解码权重和偏置; f 和 g 分别表示编码阶段和解码阶段的映射函数; σ_1 和 σ_2 分别表示编码和解码阶段的激活函数,常用的非线性激活函数有Tanh、Relu和Sigmoid等三种.

自编码网络的目标是通过不断训练来减小原始输入数据 x 和目标数据 $\hat{x}$ 之间的重构误差 $J(x, \hat{x})$,本文重构误差函数 $J(x, \hat{x})$ 选择均方误差损失函数,计算公式如式(3)所示.

$$J(x, \hat{x}) = \frac{1}{m} \sum_{i=1}^m \frac{1}{2} ||x - \hat{x}||^2 \quad (3)$$

自编码网络输出层的神经元个数完全等于输入层的神经元个数,网络通过将原始输入数据压缩到隐藏层,然后在输出层恢复原始数据样本,即 $\hat{x}_i = f_{w,b}(x_i) \approx x_i$.若不加任何限制,自编码网络很容易将原始输入数据直接作为输出,那么对目标特征的提取将没有任何意义,为此本文算法在误差损失函数 $J(x, \hat{x})$ 后加入正则化约束项来构造稀疏自编码网络,具体计算如公式(4~6)所示.

$$\tilde{\rho}_j = \frac{1}{m} \sum_{i=1}^m [k_j(x^{(i)})] \quad (4)$$

$$KL(\rho || \tilde{\rho}_j) = \rho \log \frac{\rho}{\tilde{\rho}_j} + (1 - \rho) \log \frac{1 - \rho}{1 - \tilde{\rho}_j} \quad (5)$$

$$J_{\text{sparse}}(x, \hat{x}) = J(x, \hat{x}) + \omega \sum_{j=1}^m KL(\rho || \tilde{\rho}_j) \quad (6)$$

针对深度神经网络易出现过拟合问题,通过在加入正则化约束项的误差函数后加入权重衰减项缓解过拟合.稀疏自编码网络(Sparse Autoencoder, SAE)代价函数计算如式(7)所示.

$$J_{\text{SAE}}(x, \hat{x}) = J(x, \hat{x}) + \omega \sum_{j=1}^m KL(\rho || \tilde{\rho}_j) + \frac{\lambda}{2} \sum_l \sum_{i=1}^{s_{l-1}} \sum_{j=1}^{s_l} (\omega_{ij}^{(l)}) \quad (7)$$

式中, k_j 表示隐藏层神经元 j 相对于输入 x 的激活度, m 表示训练样本的个数, ω 为正则约束项权重,约束隐藏层激活值的稀疏程度, $\tilde{\rho}_j$ 表示隐藏层神经元 j 的平均激活度, ρ 为稀疏因子, $KL(\rho || \tilde{\rho}_j)$ 为正则约束项, λ 表示相邻两层 s_{l-1} 和 s_l 的权重约束项因子.

2.2 降噪自动编码器网络

降噪自编码器(Denoising Autoencoder, DA)是Vincent等人于2008年提出的一种用于缓解传

统自编码网络模型容易过拟合的深度神经网络模型。通过在输入数据中加入部分干扰数据来训练自编码网络模型,提取具有更高鲁棒性的特征,进而提高模型的泛化能力。

为增强模型提取特征时具有更高的鲁棒性,自编码网络在编码前通过随机映射函数 H 将原始输入数据 x 随机重置为 0,构造干扰数据 $\tilde{x} \sim H(x, 0)$;然后利用干扰数据 $\tilde{x}$ 计算编码数据和解码数据,即 $y = f(x) = \sigma_1(xW_1 + b_1)$ 和 $\hat{x} = g(y) = \sigma_2(yW_2 + b_2)$;最后通过计算原始输入数据 x 与目标数据 $\hat{x}$ 之间的重构误差 $J(x, \hat{x})$,并使用梯度下降算法进行训练来减少重构误差 $J(x, \hat{x})$,使模型尽可能重构出原始输入数据,即 $J(x, \hat{x}) \approx 0$,从而增强模型提取特征时的鲁棒性和模型的泛化能力。

2.3 深度自编码网络

传统神经网络模型训练时,采用反向传播的方法进行模型训练,导致传播过程中因误差更新信号逐层减弱,使得模型无法实时更新。为此本文通过将多个按照前向传播方式进行训练的单层自编码器叠加构造深度自编码网络,并使用逐层贪婪训练的方法克服因网络层数增加而导致“梯度消失”的问题,即上一隐含层训练输出的结果作为下一层的输入。

2.3.1 预训练

深度自编码网络是深度神经网络的一种特殊模型,为克服深度神经网络模型训练时容易出现“梯度消失”问题,本文采用逐层贪婪训练算法对每层网络进行训练。即首先训练第一个自编码网络,使原始输入数据与目标数据之间的重构误差最小,然后将第一个自编码网络的输出作为下一个自编码网络的输入,不断优化重构误差,直到各层自编码网络训练完毕。

Step1:使用单层自编码器训练输入层到 H_1 层的参数,待训练完毕,去除 H_1 层的解码层,只保留输入层到隐藏层的编码阶段;

Step2:将 H_1 层神经元的激活值作为 H_2 层的输入,待训练完毕,去除 H_2 层的解码层;

Step3:将 H_2 层神经元的激活值作为 H_3 层的输入,待训练完毕,去除 H_3 层的解码层;

Step4:重复上述步骤,直到全部自编码器训练完毕。

2.3.2 参数微调

参数微调是利用有监督学习方法对网络各层参数进行微调,使得原始输入数据与目标输出之间的

重构误差最小化,本文采用 BP 算法实现网络各层参数的微调。有监督训练流程如图 1 所示。

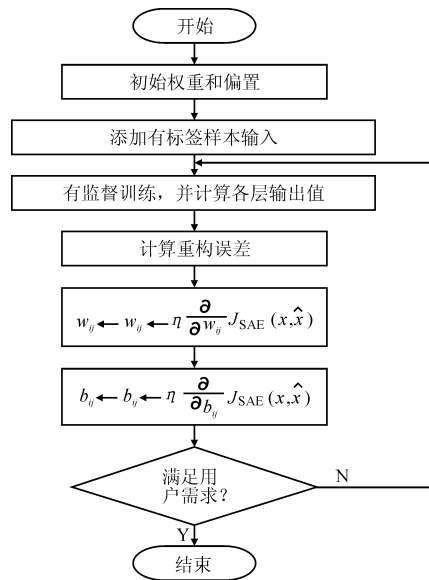


图 1 有监督训练流程

2.4 决策树算法

2.4.1 特征匹配函数

根据提取的域名字符统计特征,构造恶意域名检测特征匹配函数^[8]。特征匹配函数如公式(8)所示。

$$y(x) = \frac{f(x_i)}{g(y_j)} \quad (8)$$

式中, $y(x)$ 表示待测域名 x 是否为恶意域名的标记, $f(x_i)$ 表示待测域名 x 中第 i 个字符统计特征属性, $g(y_j)$ 表示特征集合中第 j 个域名字符统计特征属性。当 $y(x)$ 取值等于 1 时,表示待测域名 x 中第 i 个字符统计特征属性与特征集合中第 j 个字符统计特征属性匹配;否则表示特征不匹配。

2.4.2 决策树构造

根据提取的域名字符统计特征,构造本文恶意域名检测算法的决策树。决策树构造流程如算法 1 所示。

算法 1: 决策树构造

- 1 Inputs: Normal domain names $D_1 = \{d_1, \dots, d_n\}$;
Malicious domain names $D_2 = \{d'_1, \dots, d'_m\}$;
- 2 Initialize: Leaf node: $\Pi = \{N_1, \dots, N_l\}$;
An empty decision tree $DT = \{\}$;
Features attribute set $U = \{u_1, u_2, \dots, u_k\}$;
- 3 for $i = 1; (m+n)$
- 4 $U_1 = \text{extract}(\text{Malicious domain names})$;
- 5 $U_2 = \text{extract}(\text{Normal domain names})$;
- 6 $U = U_1 \cup U_2$;
- 7 $R(N_i)$ matches U ;
- 8 end

9 Output: Decision tree DT

2.4.3 恶意域名识别

在恶意域名识别过程中,按照待测域名字符统计特征与特征集合之间的匹配度给出是否为恶意域名的判断. 恶意域名检测流程如算法 2 所示.

算法 2: 恶意域名检测

```
1 Inputs: Domain name to be tested  $D=\{x_1,\cdots,x_n\}$ ;  
    $U=\{u_1,u_2,\cdots,u_k\}$ ;  
2  $U'$ =extract( $x_1,\cdots,x_n$ );  
3 for  $i=1:n$   
4   if  $R(x_i)_j=f(x_i)_j/U_l=1$  then  
5     return  $x_i$  is malicious;  
6   else  
7     return  $x_i$  is potential malicious;  
8   if  $U'(x_i)_1,U'(x_i)_2,\cdots,U'(x_i)_m\notin U_1$  then  
9     return  $y(x_i)$  is normal;  
10  end  
11 end  
12 end  
13 end  
14 Output: Malicious or normal domain names
```

3 实验与分析

3.1 实验环境

基于深度自编码网络和决策树的恶意域名检测实验环境为 64 位 windows 10,CPU 为 AMD A12-9700 2. 5GHz, 8GB 内存, GPU 为 AMD R8 M435DX,开发语言为 Python 3. 5. 2,测试平台为 Jupyter Notebook.

为验证本文算法的性能,从 Alexa^[9]、Malware domain list^[10] 和 Anquan organization^[11] 等文件中收集并整理 180 000 条域名,其中 70%的域名作为训练数据,剩余 30%作为测试数据. 同时使用准确率(Accuracy Rate, AR),假阴性率(False Negative Rate, FNR)、假阳性率(False Positive Rate, FPR)和精确率(Precision Rate, PR)来评价模型的性能,具体计算如公式(9)~(12)所示.

$$AR = \frac{N_{m \rightarrow m} + N_{n \rightarrow n}}{N_{m \rightarrow m} + N_{n \rightarrow n} + N_{m \rightarrow n} + N_{n \rightarrow m}} \quad (9)$$

$$FNR = \frac{N_{m \rightarrow n}}{N_{m \rightarrow n} + N_{m \rightarrow m}} \quad (10)$$

$$FPR = \frac{N_{n \rightarrow m}}{N_{n \rightarrow m} + N_{n \rightarrow n}} \quad (11)$$

$$PR = \frac{N_{m \rightarrow m}}{N_{m \rightarrow m} + N_{n \rightarrow m}} \quad (12)$$

式中, $N_{n \rightarrow n}$ 表示被算法准确检测出的合法域名总数; $N_{m \rightarrow n}$ 表示将恶意域名误报为合法域名的个数; $N_{n \rightarrow m}$ 表示将合法域名误报为恶意域名的个数; $N_{m \rightarrow m}$ 表示被算法准确检测出的恶意域名总数.

在恶意域名检测实验时,首先进行参数的调优,目前针对神经网络参数的选择只能在先验知识的基础上进行多次实验,通过对比分析不同参数下实验的结果,确定模型的最终参数. 具体参数如表 1 所示.

表 1 超参数设置

参数	数值
网络结构	128-50-20-5
Learning Rate	0. 001
正则约束项权重 ω	0. 17
ρ 稀疏因子	0. 02
训练迭代次数	80
微调次数	35
相邻两层权重约束项因子 λ	0. 15

3.2 实验及结果分析

图 2 和图 3 分别给出了训练集与测试集上准确率和损失值与训练次数之间的对应关系.

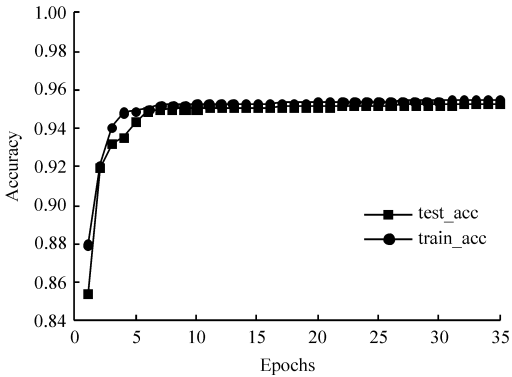


图 2 Accuracy 曲线

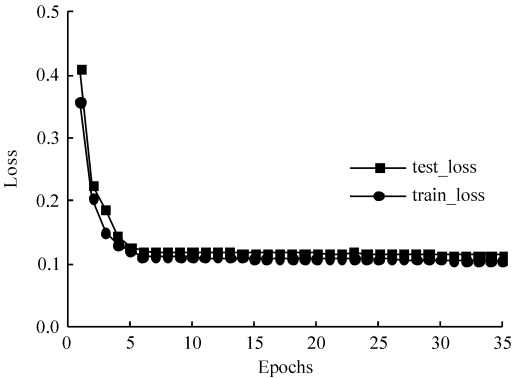


图 3 Loss 曲线

由图 2 可知,在训练与测试初期,模型的准确率随着训练步数的增加而快速上升,且训练准确率高于测试准确率;由图 3 可知,在训练初期模型的 loss 值随着训练步数的增加快速下降,且训练 loss 值低于测试 loss 值,表明该模型具有较好的训练与测试效果.

3.3 性能比较

本文提出并实现了基于深度自编码网络和决策树的恶意域名检测模型,为验证模型检测恶意域名

时的性能,分别与基于 RF(Random Forest)、AB(Adaboost)、SVM(Support Vector Machine)和 LR(Logistic Regression)等恶意域名检测模型进行比较,性能比较结果如表 2 所示。

表 2 不同模型的性能比较

	AR/%	PR/%	FNR/%	FPR/%
RF	93.41	93.08	3.78	1.49
AB	93.13	92.37	4.41	4.85
SVM	93.90	92.72	6.08	5.44
LR	94.07	93.12	3.29	1.17
DAE-DT	95.21	94.17	2.41	3.63

由表 2 可知,在准确率 AR 方面,基于 DAE-DT 的恶意域名检测模型与 SVM、AB 和 RF 模型相比优势明显,具有较好的检测性能;在检测精准率方面,本文域名检测模型与 AB 和 SVM 相比优势明显;在假阴性率方面,本文域名检测模型与 LR、SVM、AB 和 RF 等恶意域名检测模型相比优势明显,能够较好的检测出恶意域名;在假阳性率方面,本文域名检测模型与 SVM 和 AB 模型相比优势明显。

图 4 给出了 5 种模型的 ROC(Receiver Operating Characteristic Curve)曲线,其中横坐标表示 $FPR=FP/(FP+TN)$,纵坐标表示 $TPR=TP/(TP+FN)$ 。基于深度自编码网络和决策树的恶意域名检测模型的 ROC 曲线一直位于其他域名检测模型 ROC 曲线的上方,曲线下方的面积(Area Under Curve, AUC)相比其他四种域名检测模型有了 3.20% ~ 8.99% 的提升,表明本文恶意域名检测模型具有更好的综合性能。

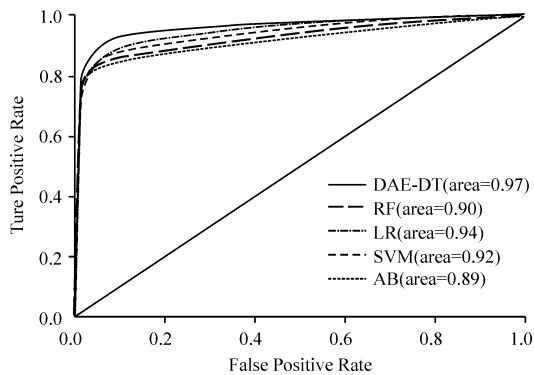


图 4 ROC 曲线

4 结束语

针对目前恶意域名检测方法特征提取过程复杂和检测准确率不高的问题,在已有研究成果的基础上,通过叠加单层自编码器构造深度自编网络,并将

其应用到恶意域名检测研究中,并传统恶意域名检测模型进行对比,实验结果表明,本文恶意域名检测模型表现良好,具有较好的实用价值。

参考文献:

- [1] 网络安全与信息动态周报. 第 30 期互联网安全威胁报告 [EB/OL]. [2019-08-10]. <http://www.cert.org.cn/publish/main/44/index.html>.
- [2] ZHAO H, CHANG Z B, BAO G B, et al. Malicious domain names detection algorithm based on N-Gram [J]. Journal of Computer Network and Communications, 2019(2):1-9.
- [3] XU C, SHEN J, DU X. Detection method of domain names generated by DGAs based on semantic representation and deep neural network[J]. Computers & Security, 2019(85): 77-88.
- [4] YANG P, ZHAO G Z, ZENG P. Phishing website detection based on multidimensional features driven by deep learning [J]. IEEE Access, 2019 (7): 15196-15209.
- [5] YU B, JIE P, DANIEL G, et al. Weakly supervised deep learning for the detection of domain generation algorithms[J]. IEEE Access, 2019(7): 51542-51556.
- [6] HUANG D, XU K, PEI J. Malicious url detection by dynamically mining patterns without pre-defined elements [J]. World Wide Web, 2014, 17 (6): 1375-1394.
- [7] 丁红卫, 万良, 龙廷艳. 深度自编码网络在入侵检测中的应用研究[J]. 哈尔滨工业大学学报, 2019, 51 (5): 185-194.
- [8] 朱俚治. 基于决策树的病毒检测算法[J]. 计算机与数字工程, 2015, 43(6): 1069-1072.
- [9] Alexa Top 1M [EB/OL]. [2019-05-18]. <https://scotthelme.co.uk/alexa-top-1-million-analysis-february-2018/>.
- [10] Malware Domain List [EB/OL]. [2019-05-18]. <http://www.malware-domainlist.com/mdl.php>.
- [11] Anquan Organization [EB/OL]. [2019-05-18]. <https://www.anquan.org/cert/>.

作者简介:

赵宏男, (1971-), 博士, 教授. 研究方向为并行与分布式处理、自然语言处理、深度学习.

常兆斌(通讯作者)男, (1995-), 硕士研究生. 研究方向为深度学习、空间网络安全、自然语言处理.

E-mail: 1510998508@qq.com.

王伟杰女, (1994-), 博士研究生. 研究方向为深度学习和语音识别.