MalMKNet:一种用于恶意代码分类的多尺度卷积神经网络

张丹丹,宋亚飞,刘 曙
(空军工程大学防空反导学院,陕西西安 710051)

摘 要: 对未知恶意代码及其变种进行快速准确地识别,是对恶意攻击行为进行有效防范的前提和基础.但随着恶意代码变种的急剧增加,人工更新样本数据库的效率越来越差,仅仅依据延时的数据库信息,传统的识别方法难以有效捕获经过混淆方法操作的样本特征信息.针对上述问题,本文设计了一种基于灰度图像处理的深度学习模型MalMKNet(Multi-scale Kernel Network for Malware),建立了一种多尺度卷积核混合的卷积神经网络(Convolutional Neural Network,CNN)架构,以提高恶意代码识别能力.该模型运用具有捷径(shortcut)结构的深度大内核卷积和标准小内核卷积相结合的混合卷积核(Mixed Kernels,MK)模块,以提高模型准确率;在此基础上,通过多尺度内核融合(Multi-scale Kernel Fusion,MKF),以降低模型参数量;再结合特征重组(feature shuffle)操作,实现优化特征通信,在不增加模型参数量的前提下提升了分类精度.实验结果表明,MalMKNet在恶意代码家族分类准确率方面优于其他基于深度学习的分类方法,准确率达到了99.35%.

关键词: 恶意代码识别;卷积神经网络;深度学习;图像处理;大卷积核;轻量化模型

基金项目: 国家自然科学基金(No.61806219,No.61703426,No.61876189);陕西省自然科学基金(No.2021JM-226);陕西省高校科协青年人才托举计划(No.20190108,No.20220106);陕西省创新能力支撑计划(No.2020KJXX-065)

中图分类号: TP309.5 **文献标识码:** A **文章编号:** 0372-2112(2023)05-1359-11

电子学报URL:http://www.ejournal.org.cn **DOI:**10.12263/DZXB.20221069

MalMKNet: A Multi-Scale Convolutional Neural Network Used for Malware Classification

ZHANG Dan-dan, SONG Ya-fei, LIU Shu
(*Institute of Air Defense and Anti-missile, Air Force Engineering University, Xi'an, Shaanxi* 710051, *China*)

Abstract: Rapid and accurate identification of unknown malware and its variants is the premise and basis for the effective prevention of malicious attacks. However, with the rapid increase of malware variants, the efficiency of manual updating of the sample database is getting worse and worse. It is difficult for the traditional identification method to effectively capture the sample feature information operated by the confusion method only based on the delayed database information. To address the above problems, this paper proposes a deep learning model based on grayscale image processing, MalMKNet (Multi-scale Kernel Network for Malware), a convolutional neural network (CNN) architecture using multi-scale convolution kernel mixing action to improve malware detection capabilities. The mixed kernels (MK) module combining deep large kernel convolution and standard small kernel convolution with shortcut structure is proposed to improve the model accuracy, and then we proposed multi-scale kernel fusion (MKF) to reduce the number of parameters. The feature shuffle (FS) is proposed to improve the classification accuracy without increasing the number of parameters. Experimental results show that MalMKNet outperforms the state-of-the-art methods in terms of malware family classification accuracy which achieves 99.35%.

Key words: malware detection; convolutional neural network; deep learning; image processing; large kernels; lightweight model

Foundation Item(s): National Natural Science Foundation of China (No.61806219, No.61703426, No.61876189); Natural Science Foundation of Shaanxi Province (No.2021JM-226); Young Talent Fund of Association for Science and

收稿日期:2022-09-20;修回日期:2022-11-28;责任编辑:覃怀银

Technology in Shaanxi, China (No.20190108, No.20220106); Innovation Capability Support Program of Shaanxi (No.2020KJXX-065)

1 引言

恶意代码是指在计算机系统中通过非授权操作来实施破坏或窃取信息的代码,以达到影响数字系统完整性、机密性和功能的目的[1].《2020年中国网络安全报告》[2]显示:2020年瑞星"云安全"系统共截获病毒样本总量1.19亿个,病毒感染次数2.59亿次,勒索软件、挖矿病毒持续活跃. 恶意代码编写者设计了大量混淆方法来增强恶意代码的隐蔽性,如死代码插入、代码转置、重排等技术,以创建当前恶意代码系列的变体来逃避检测[3]. 由Kelihos、Storm等多种恶意代码变体之间的相似性可知它们是由相同的编码人员开发设计的[1]. Greengard等[4]研究表明,大多恶意代码都是由已知恶意代码变种而来,其中差异性不到2%. 因此对恶意代码进行准确分类能够便于研究人员更好地把握恶意代码变种规律,有针对性地进行防范.

当前恶意代码检测技术主要基于恶意代码的静态和动态特征进行分析. Venkatraman等[5]指出静态分析可以迅速捕获到与结构属性相关的信息,动态分析可以有效地利用运行时的信息. 但这些技术大多依赖于特征工程工作或领域知识来构建特征数据库. 有研究人员将目光投向了图像可视化方法. Natraj等[6]将二进制可执行文件的结构转换为灰度图像纹理,通过分析纹理相似性来进行恶意代码分类. 结果表明,纹理分析耗时更少且分类性能也更好. Makandar[7]指出图像可视化方法虽然解决了恶意代码检测技术普适性问题,但仍需要较高的计算成本来提取复杂的纹理特征. 深度学习已应用于语音和图像识别等相关领域[8,9]. 文献[10,11]将数据均衡方法及多层感知器与卷积神经网络相结合,以解决过拟合问题,然而在准确性方面不尽人意. 文献[12]通过使用相似性挖掘和成本敏感性的深度学习架构对模糊恶意代码的多种特征进行分析学习. 文献[13,14]将恶意代码转化后的灰度图像输入先进的神经网络VGG-16与ResNeXt进行分类. 这些基于卷积神经网络的方法大多都是通过堆叠较小的卷积核来增大感受野,每个输出所包含信息的范围较小. 最近,已有研究表明在计算机视觉任务中,采用大卷积核使得网络具有较大的感受野,模型效果较好[15]. 但使用大卷积核会显著增加模型的参数量,因此需要寻求模型性能和参数量的折中.

2 MalMKNet模型

2.1 数据预处理模块

在原有恶意代码基础上使用混淆方法创建的变体与原二进制文件结构非常相似,将恶意代码二进制文件可视化为图像,可以检测出发生的微小变化,同时保留属于同一家族样本的全局结构. 本文采用文献[6]中给出的方法,将恶意代码二进制文件转换为灰度图,可视化过程如图1所示. 将二进制文件中的每8位看作是0~255范围内的无符号整数向量,即为一个像素. 像素值计算如下:

$$P_{\text{bin}}=b_0\times2^0+b_1\times2^1+b_2\times2^2+b_3\times2^3+b_4\times2^4+b_5\times2^5+b_6\times2^6+b_7\times2^7 \tag{1}$$

像素值与灰度值取值区间相对应,得到数值位于[0,255]的数组矩阵,最末端数据不足处以0填充. 图像的宽度是固定的,高度根据文件大小自适应. 如表1所示,文献[6]中给出了不同文件大小的推荐图像宽度.

由恶意代码二进制文件转换而成的灰度图像尺寸并不固定,若采用上表给定的图像尺寸,部分图像在输入时,由于长宽比例不一致,将被强制缩放导致图像失真. 为避免因图像失真而造成的模型分类精度下降问题,本文在将样本图像输入模型前,在不改变原始图像长宽比的前提下对短边进行缩放,然后对长边进行中心裁剪,得到64×64大小的正方形图像.

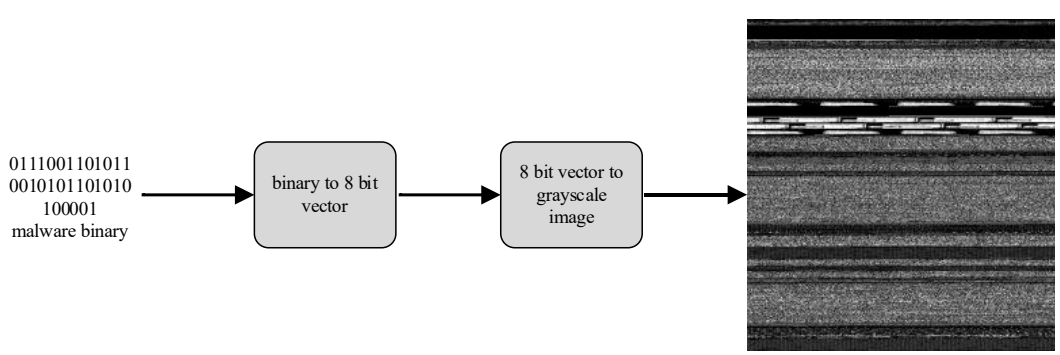


图1 恶意代码二进制文件可视化过程

表1　不同文件大小的图像宽度

文件大小	图像宽度
<10 KB	32
10 KB~30 KB	64
30 KB~60 KB	128
60 KB~100 KB	256
100 KB~200 KB	384
200 KB~500 KB	512
500 KB~1 000 KB	768
>1 000 KB	1 024

2.2　CNN模块

本文提出的模型结构以深度卷积作为网络的基本组件. 深度卷积通过更宽的特征映射提取图像的高维特征,同时能够大幅减少参数,这为较小的模型带来了更多的好处. 本文的每个深度卷积或标准卷积模块都包括三层:(1)卷积层、(2)批量归一化(Batch Normalization, BN)层[16]和(3)激活层. 卷积层对输入图像进行特征提取、特征映射、权重共享、局部连接等操作. 卷积层可以减小图像的大小,并进一步降低后续操作的计算成本. 卷积运算如式(2)所示.

$$\boldsymbol{v}_j=\boldsymbol{X}*\omega_j=\sum_{k=1}^{n}(\boldsymbol{X}_k*\omega_{j,k}) \tag{2}$$

其中,n是输入矩阵的个数,$\boldsymbol{X}_k$是第k个输入矩阵,ω_j是滤波器j的卷积核,$\omega_{j,k}$是卷积核ω_j的第k个子卷积核矩阵,*表示卷积运算.

BN层可以加快模型训练时的收敛速度,降低网络的学习难度,使得模型训练过程更加稳定,避免梯度爆炸或者梯度消失. BN如式(3)所示.

$$\begin{aligned}\boldsymbol{O}_j&=\frac{\boldsymbol{v}_j-\mathrm{E}[\boldsymbol{v}]}{\sqrt{\mathrm{Var}[\boldsymbol{v}]+\varepsilon}}\times\boldsymbol{\gamma}_j+\boldsymbol{\beta}_j\\&=\frac{\sum_{k=1}^{n}(\boldsymbol{X}_k*\omega_{j,k})-\boldsymbol{\mu}_j}{\boldsymbol{\sigma}_j}\times\boldsymbol{\gamma}_j+\boldsymbol{\beta}_j\\&=\sum_{k=1}^{n}(\boldsymbol{X}_k*\frac{\boldsymbol{\gamma}_j}{\boldsymbol{\sigma}_j}\omega_{j,k})-\frac{\boldsymbol{\mu}_j\boldsymbol{\gamma}_j}{\boldsymbol{\sigma}_j}+\boldsymbol{\beta}_j\\&=\boldsymbol{X}\otimes\omega_j\end{aligned} \tag{3}$$

其中,$\mathrm{E}[\boldsymbol{v}]$和$\mathrm{Var}[\boldsymbol{v}]$分别为向量$\boldsymbol{v}$中元素的均值和方差,$\boldsymbol{\mu}_j=\mathrm{E}[\boldsymbol{v}],\boldsymbol{\sigma}_j=\sqrt{\mathrm{Var}[\boldsymbol{v}]+\varepsilon}$,$\boldsymbol{\gamma}$、$\boldsymbol{\beta}$是学习的比例系数和偏置项,都是可训练参数,⊗表示带BN的卷积运算.

激活层对卷积层的输出执行非线性映射. 平滑的Mish激活函数允许更好的信息深入神经网络,从而得到更好的准确性和泛化能力[17]. Mish激活函数如式(4)所示.

$$\mathrm{Mish}(\boldsymbol{x})=\boldsymbol{x}\times\tanh(\ln(1+\mathrm{e}^{\boldsymbol{x}})) \tag{4}$$

2.3　MK模块

MK模块如图2所示,每个MK块由捷径和具有大内核的深度卷积构成. 除了深度方向大内核外,其他内容包括1×1卷积和批量归一化. 对于卷积核尺寸较大的网络模型,捷径使模型成为由具有不同结构感受野的模块组成的集合,在受益于更大感受野的同时不会失去捕获局部细节的能力. 除了大内核卷积提供足够的感受野和聚集空间信息的能力外,模型的表征能力也与深度密切相关. 为了获取更多的非线性特性和跨通道的信息通信,模型在具有大内核的深度卷积层前使用1×1卷积层来增加深度,在1×1卷积层之前使用批量归一化,使之完全融合到卷积层中进行有效推理. 每个具有大内核的深度卷积层都使用一个小内核进行多尺度内核融合,增加特征多样性,避免因直接使用大卷积核而造成模型精度下降的问题.

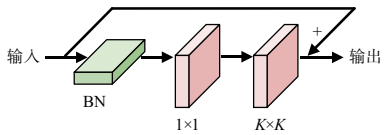


图2　MK模块的结构

在MK模块中,模型使用了具有大内核的深度卷积提高分类性能,为解决直接使用大核卷积造成的参数量增加问题,引入了多尺度内核融合. 融合了小内核的大卷积核也能够捕获更多细微特征,以此提高模型性能,增加的额外局部性也使得网络在小数据集上更容易进行优化,有助于解决优化问题. 图3展示了多尺度内核融合的过程.

(1) BN融合:卷积的同质性允许批量归一化等效地融合到具有附加偏置的卷积层中. 从式(3)中可以看出,对于每个分支,若需对卷积核ω_j产生相同的输出,构造$-\frac{\boldsymbol{\mu}_j\boldsymbol{\gamma}_j}{\boldsymbol{\sigma}_j}+\boldsymbol{\beta}_j$作为$\frac{\boldsymbol{\gamma}_j}{\boldsymbol{\sigma}_j}\omega_j$的附加偏置项即可.

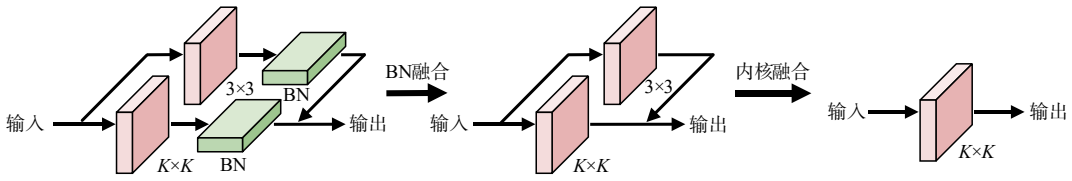


图3　多尺度内核融合过程

（2）内核融合：首先将两个BN分支合并到卷积层中．这种合并是通过构建原始结构的网络并使用融合权重进行初始化来实现的，对于每个滤波器j，设ω_j是融合后的深度大卷积核，$\boldsymbol{b}_j$是获得的偏置项，$\bar{\omega}_j$和$\hat{\omega}_j$分别是3×3卷积和$K\times K$卷积层对应滤波器的核，则有

$$\omega_j=\frac{\bar{\gamma}_j}{\bar{\sigma}_j}\bar{\omega}_j\oplus\frac{\hat{\gamma}_j}{\hat{\sigma}_j}\hat{\omega}_j \tag{5}$$

$$\boldsymbol{b}_j=-\frac{\bar{\mu}_j\bar{\gamma}_j}{\bar{\sigma}_j}-\frac{\hat{\mu}_j\hat{\gamma}_j}{\hat{\sigma}_j}+\bar{\boldsymbol{\beta}}_j+\hat{\boldsymbol{\beta}}_j \tag{6}$$

$$\bar{\boldsymbol{O}}_j=\sum_{k=1}^{n}\left(\boldsymbol{X}_k*\frac{\bar{\gamma}_j}{\bar{\sigma}_j}\bar{\omega}_{j,k}\right)-\frac{\bar{\mu}_j\bar{\gamma}_j}{\bar{\sigma}_j}+\bar{\boldsymbol{\beta}}_j=\boldsymbol{X}\otimes\bar{\omega}_j \tag{7}$$

$$\hat{\boldsymbol{O}}_j=\sum_{k=1}^{n}\left(\boldsymbol{X}_k*\frac{\hat{\gamma}_j}{\hat{\sigma}_j}\hat{\omega}_{j,k}\right)-\frac{\hat{\mu}_j\hat{\gamma}_j}{\hat{\sigma}_j}+\hat{\boldsymbol{\beta}}_j=\boldsymbol{X}\otimes\hat{\omega}_j \tag{8}$$

其中，⊕是对应位置的核参数逐元素进行相加，也就是将较小的内核“融合”到较大的内核中．对于任意滤波器，都能证明式(9)成立．

$$\begin{aligned}\bar{\boldsymbol{O}}_j+\hat{\boldsymbol{O}}_j&=\boldsymbol{X}\otimes\bar{\omega}_j+\boldsymbol{X}\otimes\hat{\omega}_j\\&=\boldsymbol{X}\otimes(\bar{\omega}_j\oplus\hat{\omega}_j)\\&=\sum_{k=1}^{n}(\boldsymbol{X}_k*\omega_{j,k})+\boldsymbol{b}_j\end{aligned} \tag{9}$$

其中，$\bar{\boldsymbol{O}}_j$和$\hat{\boldsymbol{O}}_j$分别是3×3卷积和原始$K\times K$卷积分支的输出．

在训练阶段，首先构建一个与大层平行的3×3卷积分支，然后在BN层后将其输出相加，训练完成后的大内核卷积层将不再有小核，将小核和BN参数合并到大核中得到的模型与训练模型等价．多尺度内核融合强化了网络的特征提取能力，且减少了参数量，实现了网络的轻量化．

2.4 特征重组

现代卷积神经网络通常由具有相同结构的重复构建块组成，如Xception[18]和ResNeXt[19]，在构建块中引入有效的深度可分离卷积或群卷积，以表示网络在能力和计算成本之间能够达成良好的平衡．然而，多组卷积叠加在一起会产生一个副作用，即某个通道的输出仅来自输入通道的一小部分．深度卷积使得通道具有局部性，导致模型的泛化能力差，没有有效利用不同特征图在相同空间位置上的信息，影响模型的准确率．因此本文提出特征重组，旨在添加额外操作将输出特征进行组合生成新的特征图．

特征重组具体操作如下，假设一个具有g个组的卷积层，其输出具有$g\times n$个通道；首先将输出通道维度拆分成矩阵(g,n)的两个维度，然后将其转置为矩阵(n,g)，再将它展平成一个维度$g\times n$的通道，作为下一层的输入．

2.5 深度学习模型

MalMKNet架构如图4所示．由一个捕获(capture)层和三个阶段(stage)层组成，在每两个阶段层之间通过过渡(transition)层进行过渡．

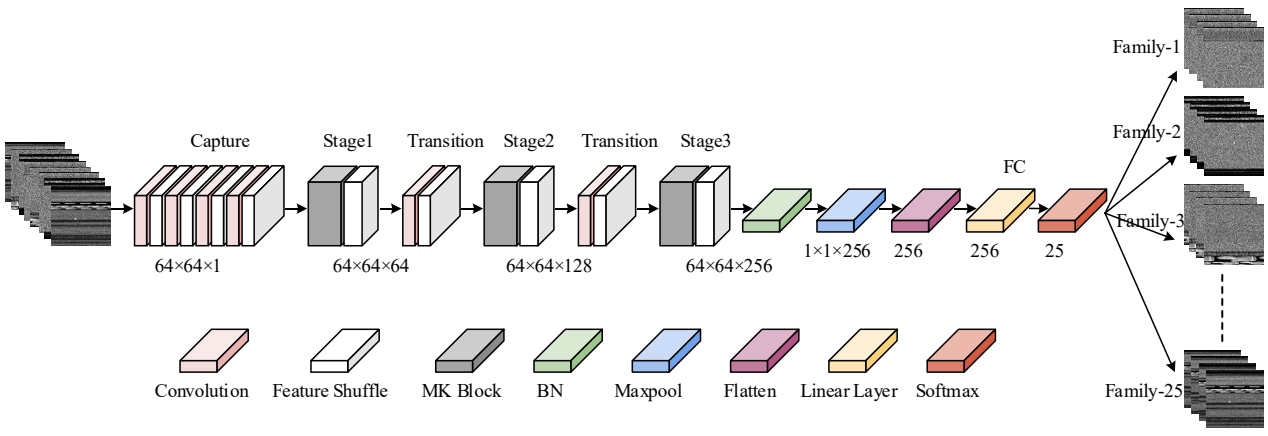


图4 MalMKNet模型结构

捕获层是开始输入神经网络的层．本文所提出的网络结构的目标是提升图像分类任务的性能，因此在开始时通过几个卷积层捕获更多图像特征．通过两个3×3深度卷积层来捕获低维度特征，两个5×5深度卷积层来提取深层特征．模型在每个深度卷积层后使用特征重组，使得后续提取到的特征更加丰富．

阶段1~3层都包含一个使用捷径的MK块，每个阶段都有两个架构超参数：通道维度C和MK块中的深度大内核尺寸K．因此，MalMKNet模型架构由$[C_1,C_2,C_3]$与$[K_1,K_2,K_3]$决定，本文提出的模型架构超参数设置$C=[64,128,256],K=[31,25,13]$.

过渡层穿插于捕获层之间，用于调整通道数以提升模型的稳定性，由3×3深度卷积和特征重组构成．

本文使用深度学习，基于上述结构训练模型．输入是经过样本预处理后大小为64×64的灰度图像，模型中的卷积层步长固定1像素，填充设置为1，确保通过卷积

层前后图像的空间维度保持一致. 输出时将各通道的特征图进行展平处理,使用Softmax分类器输出.

3 实验与分析

3.1 实验设置

本实验采用Vision Research Lab团队在2011年收集发布的Malimg恶意代码数据集进行模型训练和分类,包含了25个恶意代码家族的9 435个样本数据[6].

模型训练使用8个Nvidia特斯拉T4 GPU,各16 GB内存,操作系统Ubuntu 20.04.3,Intel(R) Xeon(R) Gold 6230R CPU@2.10 GHz×104单块CPU,512 GB计算机内存,及使用Python3.8程序代码基于神经网络框架Pytorch进行模型构建,将Malimg数据集按照9∶1的比例划分训练集和测试集,采用diffGrad优化器,学习率设置为0.002、交叉熵损失函数等. 采用上述参数,在批样本数batch size设置为64,训练轮次epochs设置为200的模型上进行训练.

为了评估模型性能,本文考虑了混淆矩阵(confusion matrix)、准确率(accuracy)、精确率(precision)、召回率(recall)和F_1分数(F_1-score),这些指标已在研究界广泛使用[20]. 为评估模型轻量化和运行时间,还考虑了参数量(parameters)和预测时间(prediction time).

3.2 超参数对比实验

将恶意代码图像规范化为不同的尺寸,并评估其性能指标. 表2给出了模型使用不同输入图像尺寸的性能评估. 从中可以看出,当恶意代码图像的尺寸从32×32增加到256×256时,参数的数量从0.21 M增加到0.53 M,检测时间从17.92 ms增加到25.11 ms. 值得注意的是,随着恶意代码图像尺寸的增加,模型分类精度最初会增加,从98.96%增加到99.35%,然后下降到98.28%,但参数量和检测时间持续增加. 分析原因,当模型的输入尺寸过大时,会发生过度过滤. 考虑到以上,模型选择64×64作为模型的输入尺寸.

表2 不同图像输入尺寸对比实验结果

输入尺寸	参数量/M	准确率/%	预测时间/ms
32×32	0.208 357	98.96	17.92
64×64	0.294 425	99.35	18.36
128×128	0.359 145	99.03	21.07
256×256	0.533 643	98.28	25.11

出于优化目的,本文对一些在分类任务中表现出色的优化器Adagrade、Adamax、Adam和NAdam进行对比实验. 在图5中,绘制了有关指标的性能图. 显然,采用diffGrad的模型收敛速度优于其他,与其他优化器相比,diffGrad优化器效率最高.

表3给出了上述优化器在实验中的比较性能,diffGrad在准确率、精确率、召回率和F_1分数方面优于其他优化器. 结果表明,模型采用diffGrad能有效执行恶意代码家族分类任务. 因此,模型采用diffGrad进行优化.

表3 不同优化器对比实验结果

优化器	准确率/%	精确率/%	召回率/%	F_1-score/%	预测时间/ms
Adagrade	97.20	96.26	97.20	96.68	18.36
Adamax	97.67	96.71	97.67	97.13	18.36
Adam	98.32	98.38	98.32	98.30	18.36
NAdam	99.00	99.08	99.00	99.01	18.36
diffGrad	99.35	99.37	99.35	99.35	18.36

3.3 消融实验

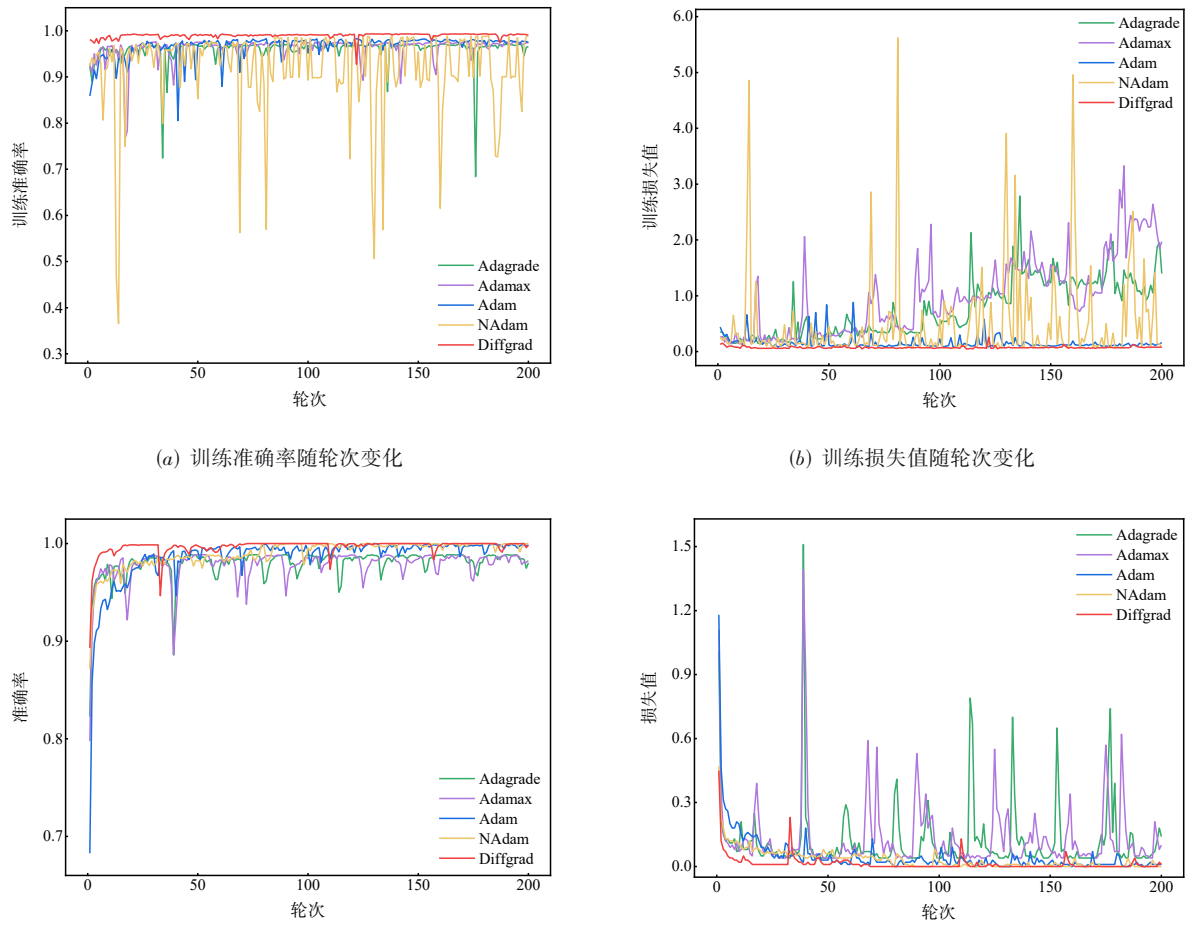
MalMKNet的核心思想在于MK块及特征重组. 在本小节中,对其中的多尺度内核融合、捷径和特征重组内容进行消融实验,表4展示了消融实验结果,混淆矩阵如图6所示.

直接使用大内核卷积虽然能提高分类精度,但这也会增加参数量. 如表4所示,使用3×3卷积融合MK模块中的大内核卷积能够提高模型分类性能,且减少了参数量. 通过这种融合方式,可大幅减少使用大内核卷积模型的运行开销.

捷径对于内核非常大的神经网络至关重要. 为了证明这一点,将所有MK模块中的捷径去掉,与标准MalMKNet模型进行对比实验. 表4表明,具有捷径的MalMKNet模型的准确性提高了2.15%. 然而,如果没有捷径,准确率将降低到仅97.20%. 实验结果表明,模型引入捷径能够缓解因梯度随着深度成倍地失去而导致

表4 消融实验结果

多尺度内核融合	捷径	特征重组	准确率/%	精确率/%	召回率/%	F_1-score/%	参数量/M
	√	√	99.03	99.06	99.03	99.03	5.413 992
√		√	97.20	96.27	97.20	96.67	0.226 484
√	√		98.24	97.27	98.24	97.72	0.294 425
√	√	√	99.35	99.37	99.35	99.35	0.294 425



(*a*) 训练准确率随轮次变化

(*b*) 训练损失值随轮次变化

(*c*) 准确率随轮次变化

(*d*) 损失值随轮次变化

图5 优化器对比实验结果

过度平滑的问题.

特征重组的目的是为卷积层启用跨组信息交换.为了公平比较,使用图4所示的整体网络架构,去掉所有的特征重组操作,由于特征重组不改变参数量及通道数,可以确保结构的复杂性保持不变.如表4所示,无论是在准确率还是在其他评估指标上,标准MalMKNet模型的性能显著优于去除特征重组的模型结构,这表明了跨组信息交换对于提升模型分类性能的重要性.然而,在MalMKNet中,各阶段通道数为[64,128,256],这表明,即使是在分组数相对较大时,特征重组也能很好的提升模型分类性能.

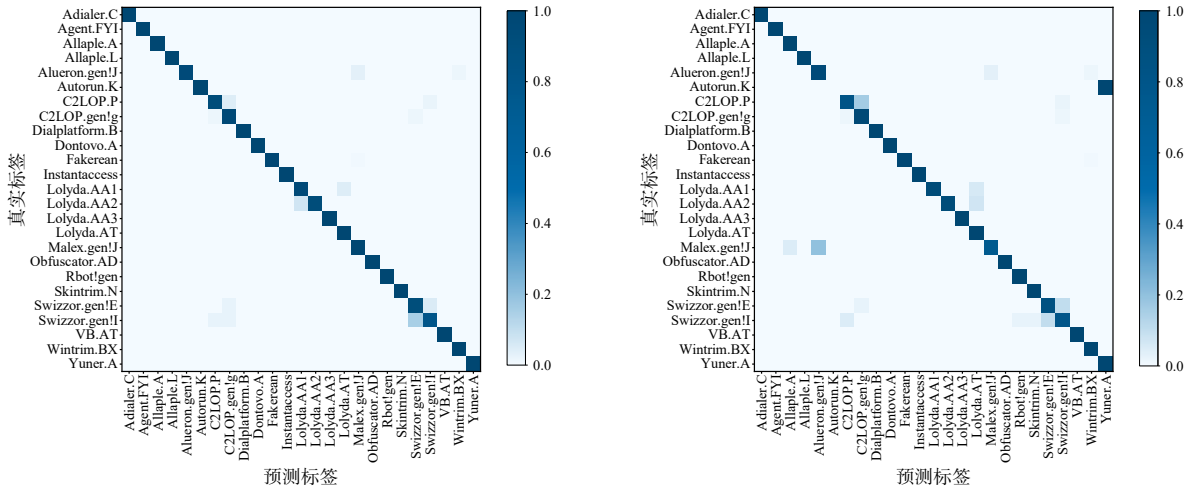
3.4 对比实验

为验证所提出MalMKNet模型的性能,所有实验在相同的环境下进行训练和测试,基于以下内容进行实验评估:(1)MalMKNet模型与其他用于图像分类任务的高级深度学习模型进行比较;(2)MalMKNet模型与已有的先进恶意代码分类技术的比较.

3.4.1 与其他模型对比实验

为了证明所提出的模型能够实现令人满意的性能,本文比较了Malimg数据集在图像分类任务中常用的一些模型上的效果,包括DenseNet、MobileNetV2、ResNet和ShuffleNet,以验证所提出的模型对恶意代码及其变种进行识别的有效性.图7展示了经过200轮训练后上述模型在恶意代码家族上的各评估指标上的表现,MalMKNet模型能够精准地对Malimg数据集上的25个家族进行分类,效果优于其他模型.表5展示了本文提出模型与其他模型性能比较的结果.显而易见,MalMKNet模型具有更高的准确率、精确率、召回率和F_1分数,对恶意代码样本的预测时间也最少.对于实现轻量化模型的目标,MalMKNet完成地非常出色,甚至优于以轻量化模型作为设计目标的MobileNetV2和ShuffleNet.

本文从模型设计的角度考虑了恶意代码之间的逻



(*a*) MalMKNet-MKF　　(*b*) MalMKNet-shortcut

(*c*) MalMKNet-FS　　(*d*) MalMKNet

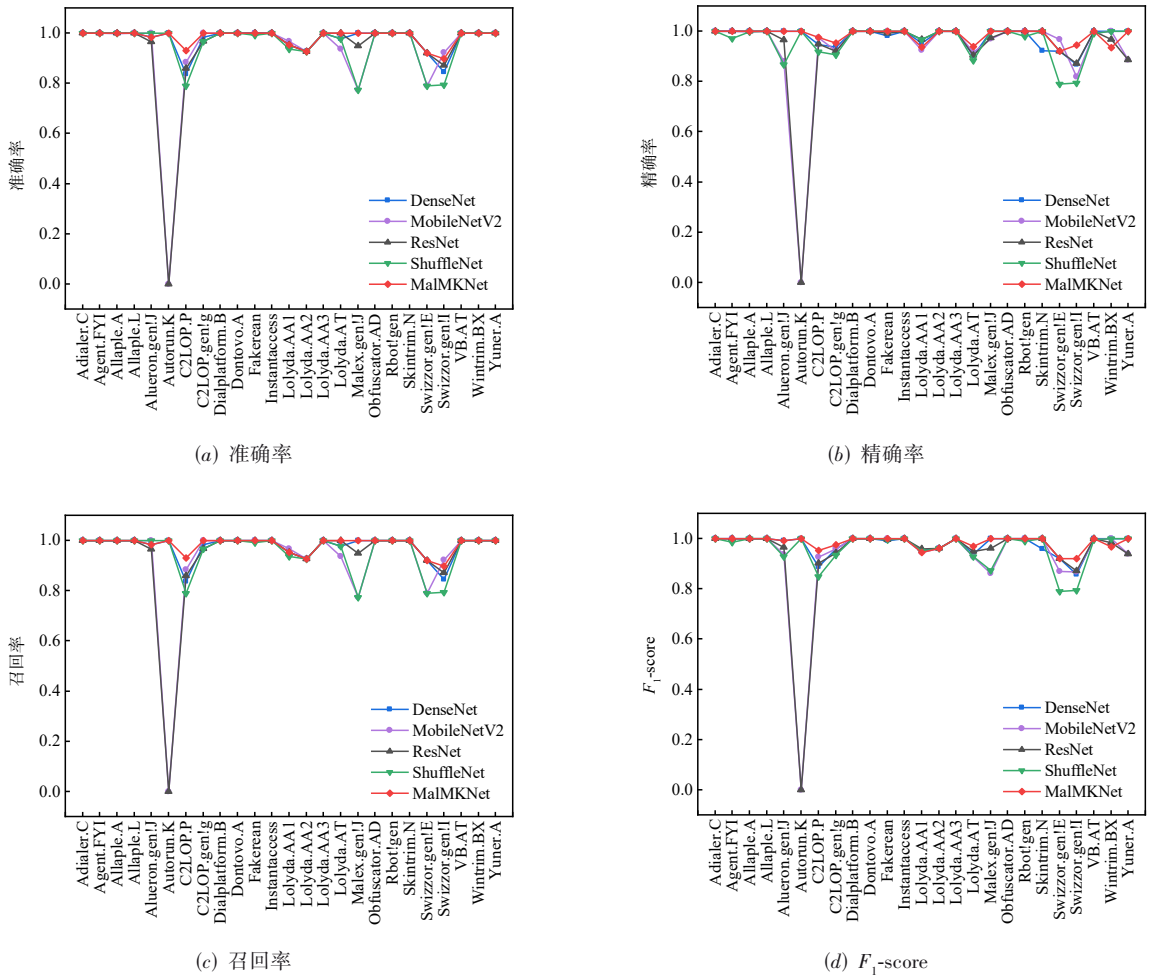
图6　模块对模型性能的影响.注:"-"表示去除该模块或该操作

辑关系,尤其是样本预处理的过程中,恶意代码变体的文件大小和图像纹理都发生了显著变化,本文提出的模型结构考虑了变化引起的差异,用大内核卷积增大感受野捕捉形状偏差来最小化代码差异的影响. 采用深度大内核卷积和标准小内核卷积相结合的方式,可以考虑到样本图像的全局信息,而不是局部特征. 对于恶意代码,代码之间的调用关系存在于全过程,而不仅仅存在于相邻的代码段中,因此网络的准确性能够得到显著地提高.

3.4.2　与其他研究方法对比实验

为了证明所提出的模型能够实现令人满意的性能,本文比较了MalMKNet模型与已有的恶意代码分类技术. 表6涵盖了在Malimg数据集上对基于图像的恶意代码分类技术的几乎所有研究,这些模型取得了显著效率.

如表6所示,模型在准确性、精确率等评估指标方面都具有更好的结果,优于所有现有研究技术,模型使用diffGrad优化器实现了99.35%的准确率,预测时间为18.36 ms,这比现有的最新研究技术要低得多. 分析原因,这与卷积的一个特性有关:将大小兼容的卷积核以相同的步幅在同一输入上操作产生的相同分辨率输出进行相加,就可以将这些核在相应位置上相加,以获得产生相同输出的等效核[32]. 模型采用多尺度内核融合卷积层的方式,使得小内核卷积层可以



(*a*) 准确率 (*b*) 精确率

(*c*) 召回率 (*d*) F_1-score

图7 MalMKNet模型在相关评估指标上的对比实验结果

表5 MalMKNet模型与其他模型的实验结果对比

模型	准确率/%	精确率/%	召回率/%	F_1-score/%	参数量/M	预测时间/ms
DenseNet	99.07	99.08	99.07	99.06	6.973 209	26.32
MobileNetV2	97.63	96.72	97.63	97.10	2.255 321	19.10
ResNet	97.92	96.95	97.92	97.40	21.291 225	21.39
ShuffleNet	98.35	98.41	98.35	98.33	0.366 985	18.96
MalMKNet	99.35	99.37	99.35	99.35	0.294 425	18.36

等价地融合到大内核卷积层中，免去了用于小内核卷积层推理的时间开销，在增加分类准确率的同时，不会引入额外的推理时间增加计算负担. 实验结果表明，所提出的MalMKNet模型具有很高的性能，在小数据集上具有显著的分类精度. 相比之下，其他论文中提出的方法大多都不考虑恶意代码的全局特征，而只关注感受野周围的关系，难以对差异较大的数据取得良好的结果.

表6　MalMKNet模型与其他研究方法的实验结果对比

作者	年份	数据集	模型	准确率/%	精确率/%	召回率/%	F_1-score/%	预测时间/ms
Nataraj[6]	2011	Malimg	KNN	97.18	—	—	—	—
Yue[21]	2017	Malimg	Vgg-verydeep-19	97.32	—	—	—	—
Zhihua[10]	2018	Malimg	GIST+KNN	91.9	92.1	91.7	—	60
Zhihua[10]	2018	Malimg	GIST+SVM	92.2	92.5	91.4	—	64
Zhihua[10]	2018	Malimg	GLCM+KNN	92.5	92.7	92.3	—	45
Zhihua[10]	2018	Malimg	GLCM+SVM	93.2	93.44	93	—	48
Zhihua[10]	2018	Malimg	IDA+DRBA	94.5	94.6	94.5	—	20
Dai[22]	2018	Malimg	GIST-Descriptor, SVM & KNN	97	—	—	—	—
Kumar[23]	2018	Malimg	CNN	98	—	—	—	—
Kalash[12]	2018	Malimg	M-CNN	98.52	—	—	—	—
Chen[24]	2018	Malimg	Inception-V1	99.25	—	—	—	—
Cui[25]	2019	Malimg	NSGA-II	97.6	97.6	88.4	—	—
Singh[26]	2019	Malimg	Deep CNN	96.08	—	—	—	—
Gibert[27]	2019	Malimg	CNN	98.48	—	—	—	—
Venkatraman[13]	2019	Malimg	CNN UniGRU	96	91.8	91.2	91.4	—
Venkatraman[13]	2019	Malimg	CNN BiGRU	96.3	91.8	91.5	91.6	—
Lo[28]	2019	Malimg	Xception	99.03	—	—	—	—
Cayir[29]	2020	Malimg	CapsNet	98.63	—	—	96.58	—
Cayir[29]	2020	Malimg	RCNF	98.72	—	—	96.61	—
Naeem[30]	2020	Malimg	DCNN	98.47	98.47	98.47	—	—
Naeem[30]	2020	Malimg	DCNN	98.79	98.79	98.79	—	—
Vasan[31]	2020	Malimg	IMCFN	98.82	98.85	98.81	98.75	810
本文作者	—	Malimg	MalMKNet	**99.35**	**99.37**	**99.35**	**99.35**	**18.36**

4　总结

本文将深度学习与恶意代码可视化相结合,根据恶意代码特性提出了一种轻量化方法,有效地解决了恶意代码识别领域普遍存在的准确度低、时间耗费高及计算开销大等问题. 通过实验对本文所提方法的性能进行了验证,结果表明,在各评估指标上该方法都优于其他先进恶意代码识别方法,尤其在准确率和参数量方面具有显著优势,易于在资源受限的设备上部署实现,具有突出的实践意义和应用价值.

参考文献

[1] SU J W, VASCONCELLOS D V, PRASAD S, et al. Lightweight classification of IoT malware based on image recognition[C]//HIRONORI K. 2018 IEEE 42nd Annual Computer Software and Applications Conference(COMPSAC). Piscataway: IEEE, 2018: 664-669.

[2] 国家互联网应急中心. 2020年中国互联网网络安全报告[R/OL]. (2021-07-21)[2022-12-29]. https://www.cert.org. cn/publish/main/46/2021/20210721130944504525772/20210721130944504525772_.html.

[3] YADAV B, TOKEKAR S. Recent innovations and comparison of deep learning techniques in malware classification: A review[J]. International Journal of Information Security Science, 2021, 9(4): 230-247.

[4] GREENGARD S. Cybersecurity gets smart[J]. Communications of the ACM, 2016, 59(5): 29-31.

[5] VENKATRAMAN S, ALAZAB M. Use of data visualisation for zero-day malware detection[J]. Security and Communication Networks, 2018, 2018: 1-13.

[6] NATARAJ L, KARTHIKEYAN S, JACOB G, et al. Malware images: Visualization and automatic classification [C]//GREGORY J. Proceedings of the 8th International Symposium on Visualization for Cyber Security. New York: ACM, 2011: 1-7.

[7] MAKANDAR A, PATROT A. Malware class recognition using image processing techniques[C]//2017 International Conference on Data Management, Analytics and Innovation (ICDMAI). Piscataway: IEEE, 2017: 76-80.

[8] XIANG Q, WANG X D, SONG Y F, et al. One-dimensional convolutional neural networks for high-resolution range profile recognition via adaptively feature recalibrating and

automatically channel pruning[J]. International Journal of Intelligent Systems, 2021, 36(1): 332-361.

[9] XIANG Q, WANG X D, LAI J, et al. Multi-scale group-fusion convolutional neural network for high-resolution range profile target recognition[J]. IET Radar, Sonar & Navigation, 2022, 16(12): 1997-2016.

[10] CUI Z H, XUE F, CAI X J, et al. Detection of malicious code variants based on deep learning[J]. IEEE Transactions on Industrial Informatics, 2018, 14(7): 3187-3196.

[11] HAMAD N, CHENG X C, FARHAN U, et al. A deep convolutional neural network stacked ensemble for malware threat classification in Internet of Things[J]. Journal of Circuits, Systems and Computers, 2022, 31(17): 1-13.

[12] KALASH M, ROCHAN M, MOHAMMED N, et al. Malware classification with deep convolutional neural networks[C]//2018 9th IFIP International Conference on New Technologies, Mobility and Security (NTMS). Piscataway: IEEE, 2018: 1-5.

[13] VENKATRAMAN S, ALAZAB M, VINAYAKUMAR R. A hybrid deep learning image-based analysis for effective malware detection[J]. Journal of Information Security and Applications, 2019, 47: 377-389.

[14] GO J H, JAN T, MOHANTY M, et al. Visualization approach for malware classification with ResNeXt[C]//2020 IEEE Congress on Evolutionary Computation (CEC). Piscataway: IEEE, 2020: 1-7.

[15] LIU S, CHEN T, CHEN X, et al. More convnets in the 2020s: Scaling up kernels beyond 51x51 using sparsity [EB/OL]. (2022-07-07) [2022-12-29]. https://arxiv. org/abs/2207.03620.

[16] IOFFE S, SZEGEDY C. Batch normalization: Accelerating deep network training by reducing internal covariate shift[C]//Proceedings of the 32nd International Conference on International Conference on Machine Learning - Volume 37. New York: ACM, 2015: 448-456.

[17] MISRA D. Mish: A self regularized non-monotonic neural activation function[EB/OL]. (2019-08-23) [2022-12-29]. https://arxiv.org/abs/1908.08681.

[18] CHOLLET F. Xception: Deep learning with depthwise separable convolutions[C]//2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR). Piscataway: IEEE, 2017: 1800-1807.

[19] XIE S N, GIRSHICK R, DOLLÁR P, et al. Aggregated residual transformations for deep neural networks[C]// 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR). Piscataway: IEEE, 2017: 5987-5995.

[20] NAMANYA A P, AWAN I U, DISSO J P, et al. Similarity hash based scoring of portable executable files for efficient malware detection in IoT[J]. Future Generation Computer Systems, 2020, 110: 824-832.

[21] YUE S. Imbalanced malware images classification: A CNN based approach[EB/OL]. (2017-08-27) [2022-12-29]. https://arxiv.org/abs/1708.08042.

[22] DAI Y, LI H, QIAN Y, et al. A malware classification method based on memory dump grayscale image[J]. Digital Investigation, 2018, 27: 30-37.

[23] KUMAR R, ZHANG X S, KHAN R U, et al. Malicious code detection based on image processing using deep learning[C]//Proceedings of the 2018 International Conference on Computing and Artificial Intelligence. New York: ACM, 2018: 81-85.

[24] CHEN L. Deep transfer learning for static malware classification[EB/OL]. (2018-12-18)[2022-12-29]. https://arxiv.org/abs/1812.07606.

[25] CUI Z, DU L, WANG P, et al. Malicious code detection based on CNNs and multi-objective algorithm[J]. Journal of Parallel and Distributed Computing, 2019, 129: 50-58.

[26] SINGH A, HANDA A, KUMAR N, et al. Malware classification using image representation[M]//Lecture Notes in Computer Science. Cham: Springer International Publishing, 2019: 75-92.

[27] GIBERT D, MATEU C, PLANES J, et al. Using convolutional neural networks for classification of malware represented as images[J]. Journal of Computer Virology and Hacking Techniques, 2019, 15(1): 15-28.

[28] LO W W, YANG X, WANG Y P. An xception convolutional neural network for malware classification with transfer learning[C]//2019 10th IFIP International Conference on New Technologies, Mobility and Security (NTMS). Piscataway: IEEE, 2019: 1-5.

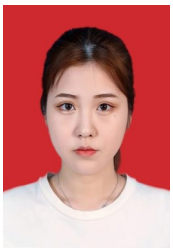
[29] ÇAYIR A, ÜNAL U, DAĞ H. Random CapsNet forest model for imbalanced malware type classification task[J]. Computers & Security, 2021, 102: 102133.

[30] NAEEM H, ULLAH F, NAEEM M R, et al. Malware detection in industrial Internet of Things based on hybrid image visualization and deep learning model[J]. Ad Hoc Networks, 2020, 105: 102154.

[31] VASAN D, ALAZAB M, WASSAN S, et al. IMCFN: Image-based malware classification using fine-tuned convolutional neural network architecture[J]. Computer Networks, 2020, 171: 107138.

[32] DING X H, GUO Y C, DING G G, et al. ACNet: strengthening the kernel skeletons for powerful CNN via asymmetric convolution blocks[C]//DAVID F. 2019 IEEE/CVF International Conference on Computer Vision (ICCV). Piscataway: IEEE, 2020: 1911-1920.

作者简介



张丹丹　女,1998年12月出生于上海市. 现为空军工程大学硕士研究生. 现为空军工程大学硕士研究生. 主要研究方向为恶意代码检测.
E-mail: afeu_ddz@163.com

宋亚飞(通讯作者)　男,1988年出生于河南汝州. 现为空军工程大学防空反导学院副教授. 主要研究方向为机器学习及其在目标识别和入侵检测等领域中的应用.
E-mail: yafei_song@163.com

刘　曙　男,1971年出生于湖南益阳. 现为空军工程大学防空反导学院副教授. 主要研究方向为网络空间信息防御和计算机与软件工程.
E-mail: liushu@163.com