基于机器学习的恶意软件检测研究进展及挑战

景鸿理[1],黄　娜[1,2],李建国[1]

(1.北京天融信科技有限公司,北京 100085;2.北京工业大学,北京 100124)

摘　要:由于恶意软件的数量日渐庞大,攻击手段不断更新,结合机器学习技术是恶意软件检测发展的一个新方向。先简要介绍恶意软件检测中的静态检测方法以及动态检测方法,总结基于机器学习的恶意软件检测一般流程,回顾了研究进展。通过使用 Ember 2017 和 Ember 2018 数据集,分析验证了结构化特征相关方法,包括随机森林(Random Forest,RF)、LightGBM、支持向量机(Support Vector Machine,SVM)、K-means 以及卷积神经网络(Convolutional Neural Network,CNN)等算法模型;使用收集的 2019 年样本集分析验证了序列化特征相关方法,包括几种常见的深度学习算法模型。计算模型以在不同测试集上的准确率、精确率、召回率以及 F1-值作为评估指标。根据实验结果分析讨论了各类方法的优缺点,着重验证分析了树模型的泛化能力,表明随着样本的不断演变,模型普遍存在退化问题,并指出进一步研究方向。

关键词:恶意软件检测;静态检测;机器学习;LightGBM;随机森林

中图分类号:TP391　　**文献标识码**:A　　**DOI**:10.19358/j.issn.2096-5133.2020.11.006

引用格式:景鸿理,黄娜,李建国. 基于机器学习的恶意软件检测研究进展及挑战[J].信息技术与网络安全,2020,39(11):38-44,68.

Research progress and challenges of malware detection method based on machine learning

Jing Hongli[1], Huang Na[1,2], Li Jianguo[1]

(1.Beijing Topsec Science & Technology Inc., Beijing 100085, China;
2.Beijing University of Technology, Beijing 100124, China)

Abstract: Due to the increasing number of malware and the updated attack means, malware detection combined with machine learning technology is a new direction of its development. Firstly, this paper introduces the static detecting methods and dynamic detecting methods of malware briefly; summarizes the general process of malware detecting methods based on machine learning, and reviews the existing methods with research progress. Using the data sets of Ember 2017 and Ember 2018, the structural feature correlation methods, including RF(Random Forest), LightGBM, SVM(Support Vector Machine), K-means and CNN(Convolutional Neural Network), are analyzed and validated,and the 2019 sample set analysis is used to validate the serialization feature correlation method, including several common deep learning algorithm models. The accuracy, precision, recall and F1_score of the trained model on different testing data sets are calculated as evaluating metrics. According to the experimental results, the advantages and disadvantages of various methods are discussed in this paper, the generalization ability of the tree model is verified and analyzed emphatically. It is shown that the model generally has degradation problem with the continuous evolution of samples, and the further research direction is pointed out at last.

Key words: malware detection; static detection of malware; machine learning; LightGBM; random forest

0 引言

恶意软件是计算机与网络领域不可避免的一项安全风险,也是安全研究者聚焦的研究热点之一。用户的隐私数据、个人信息及财产,都是恶意软件攻击的目标[1]。恶意软件自身的一些特性为检测提供了可能性和有利条件,安全研究人员提出了很多检测分析方法来遏制、打击恶意软件的发展势头。计算机技术高速发展,不仅为人们的日常生活和工

作带来了便利,也促使黑客的攻击手段和技术不断提高,使得恶意软件变得更加多元化,而且利用无线网络、局域网络、可移动设备等多种传播渠道快速传播,数量与日俱增,传统的基于特征库匹配等技术显得效率不足[2]。因此,研究者逐渐趋向于使用机器学习技术,来应对恶意软件难以预测的变种和日益庞大的数量[3]。

目前已经有许多机器学习技术和框架被研究提出,应用于恶意软件检测,起到了非常可观的效果。根据SGANDURRA D等[4]在2016年的调研,使用机器学习技术的静态检测方法准确率达到90%以上,动态检测方法准确率能够达到96%以上,经过近几年的继续发展,此类方法的性能得到了进一步提高。基于机器学习技术建立智能化检测模型,形成阻断恶意软件的一道防线,是技术突破与市场拓展的一个新方向,具有重要的研究意义和应用价值。

本文总结了基于机器学习的恶意软件检测方法的一般流程,回顾现有的研究成果;分别对结构化特征相关方法以及序列化特征相关方法进行了实验验证,结合实验结果分析讨论各类方法的适用场景以及面临的挑战,最后指出进一步研究方向。

1 基础介绍

本节简要介绍恶意软件检测方法的相关基础,从静态和动态两个角度阐述不同的特征,为分析机器学习方法作铺垫,并总结机器学习检测方法的一般流程。

1.1 方法分类

根据角度的不同,恶意软件检测方法分为静态检测和动态检测两种类型,下面介绍这两类方法,阐述存在的区别和各自的优缺点。

(1)静态检测方法

Windows系统下的可移植可执行(Portable Executable,PE)文件有统一的结构和存储数据的方式,主要包含汇编指令、图像、文本等数据,以及程序运行所需的元数据。图1展示了PE文件的结构,由一系列的头部(Header)及节(Section)组成。PE头定义程序的一般属性,如目标系统、创建时间戳、二进制代码属性、图像属性等。可选头定义PE文件程序入口点的位置、加载数据的大小以及其他有关程序的高级信息。程序入口点是逆向工程中重要的基础。节头描述了PE文件中各个节的属性,如节的名称、大小、程序为可读或可写或可执行的标记等。节则是在程序加载时被映射到内存的具体内容,节的名称可以自定义,如.text节可用于存放可执行代码,.idata节可用于列出导入的动态链接库(Dynamic Link Library,DLL)及函数,.rsrc、.data及.rdata节可用于存放图像、音频等多媒体数据。

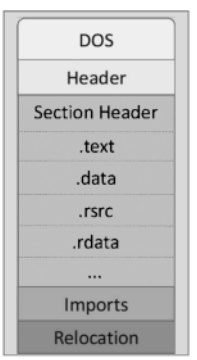


图1 Windows系统下PE文件的结构

恶意软件的静态检测方法,即根据PE文件本身所包含的信息做出分析。PE文件通常为二进制数据形式,不能直接获取以上内容信息,而且恶意软件往往采取加密、加壳等手段,需要进行解密、脱壳以及解析之后,才能获取具体内容。加密、加壳和混淆等,也是静态检测方法最主要的阻碍。与动态检测方法相比,此类方法不需要在特定环境下运行可执行文件,避免了可能产生的风险,更加易于实施。

(2)动态检测方法

恶意软件在运行时会做出各类威胁行为,包括修改文件系统(如写入设备驱动程序、更改系统配置文件)、修改注册表(如修改注册表键值、更改防火墙设置)、网络行为(如解析域名、发出HTTP请求)等。动态检测是在独立、安全的沙箱(Sandbox)环境中运行PE文件,通过行为分析来判定其是否为恶意软件。动态检测通常与可视化技术相结合,便于分析动态行为轨迹。

动态检测技术不受加壳、混淆等的限制,能够更加直接地分析PE文件行为特征,但也并非没有缺陷,动态检测的效率较低、部署困难,而且一些恶意软件能够察觉所处的运行条件或计算环境,从而逃逸检测[5]。

1.2 机器学习方法一般流程

基于机器学习的恶意代码检测方法关键在于特征和算法的选择。静态特征和动态特征都可用于机器学习,但无关特征和噪声特征会影响模型的准确性。利用数据挖掘选择数据、特征,再结合机器学习技术完成检测,是现有研究中常见的解决方案,一般分为四个步骤:数据准备,特征提取及特征选择,训练机器学习模型,获取检测结果,流程如图2所示。特征提取分为静态特征(如PE头特征、二进制内容特征等)以及动态特征(如API调用特征、系统修改特征和网络行为特征等)。然后选择一种机器学习算法,使用提取的特征集合训练模型。

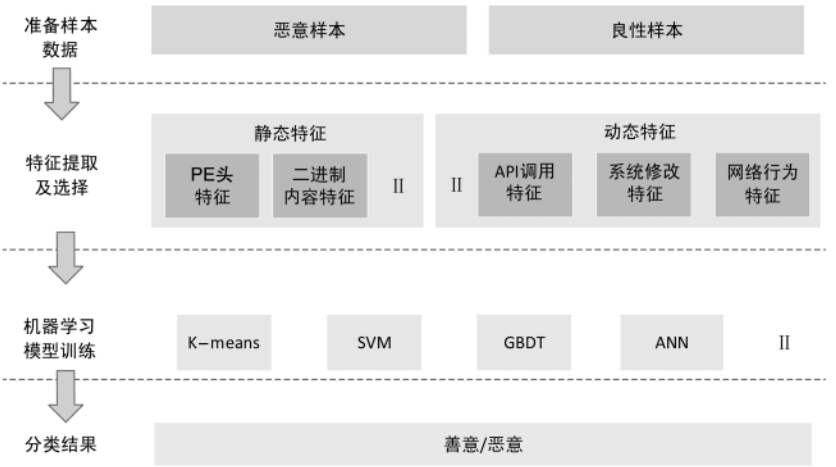


图 2 基于机器学习的恶意软件检测一般流程

2 基于机器学习的恶意软件检测

机器学习模型是通过计算目标分配到不同类别的概率来完成分类任务,应当首先检测 PE 文件是否为恶意,通过分类模型将目标文件归类为恶意或良性,然后才能对确定为恶意的 PE 文件进行族群分类。在恶意软件检测中常用的机器学习算法有距离类算法、树模型以及深度神经网络,总结如下:

(1)K-means 是一种无监督的聚类算法,将每个样本归类到距离最近的质心的类别,核心在于选取及优化质心。赵中军等[6]通过优化的 K-means 算法,快速有效地识别出恶意软件;张莹等[7]解决了传统K-means 选择初始质心不稳定的问题,提出一种基于 PSO-K-means 的恶意代码检测方法。

(2)SVM 是在数据的特征空间中寻找一个或多个最优超平面,将数据划分成不同类别,可以处理分类及回归问题。在恶意软件变种检测[8]、Android 恶意软件检测[9]、网络恶意程序检测[10]、恶意 PDF 检测[11]中有相关应用。

(3)决策树是树形结构的机器学习模型,发展出许多具体的算法,如 ID3、C4.5、Cart,以及以 Cart 为基础的随机森林以及梯度提升决策树(Gradient Boosting Decision Tree,GBDT)等。决策树算法是目前应用中准确性相对较高的一种方法,杨宏宇等[12]提出的改进随机森林算法在恶意软件检测实验中准确率达到 98%,ANDERSON H S 等[13]使用 LightGBM 算法在 Ember 2017 数据集上实验 AUC 达到了 99.91%。

(4)深度学习包括全连接神经网络、卷积神经网络(CNN)以及循环神经网络(RNN)等不同的结构,与浅层机器学习算法相比,需要学习的参数数量更多,但模型能够从输入的数据中自动学习不同层次的特征,避免了人工提取的过程。目前硬件计算能力的大幅提高,使得深度学习的普及成为可能,逐渐步入人们的视野。在恶意软件检测中,CNN、RNN 以及两者的结合应用较多,PE 文件的二进制字节内容可以直接作为深度神经网络的输入[14],也可以提取序列化的特征作为输入[15~18]。

表 1 调研了 2012 年至今的相关研究,其中 LightGBM[13]和几种深度学习方法[14-18]具有不错的表现。研究[19]、[20]为传统的相似性匹配方法,分别采用了静态特征和动态特征,通过计算相似性区分恶意软件族群。毛蔚轩等[22]为了解决恶意软件数量有限的问题,提出一种基于最小化估计风险的主动学习方法,不断地从未标记样本中寻找小风险样

表 1 相关研究总结

文献	样本数据	特征	算法	性能
文献[13]	Ember 2017	结构化静态特征,字节流特征	LightGBM	AUC 大于 0.999
文献[14]	101 万个恶意样本,100 个正常样本	字节流	CNN	AUC 98.2%
文献[15]	69 860 个恶意样本,70 140 个正常样本	结构化静态特征,字节流特征	LSTM-CNN	准确率 98.8%
文献[16]	34 995 个恶意样本,19 696 个良性样本	调用 API 序列	DNN	-
文献[17]	472 个恶意样本,371 个良性样本	行为语义特征	DNN	TPR 0.997
文献[18]	16 733 个恶意样本,1 119 个良性样本	行为特征	CNN	准确率 99.28%
文献[19]	NetSky、SdBot 族群	行为特征图	相似性匹配	匹配值大于 0.7
文献[20]	Worm、Dos、Trojan、Exploit 族群	字符矩、信息熵、相关系数等	相似性匹配	准确率大于 90%
文献[21]	-	云计算系统和网络特征	SVM	准确率大于 90%
文献[22]	7 257 个恶意样本,8 340 个正常样本	调用依赖图	主动学习(KNN;RF)	5.55%错误率
文献[23]	6 994 个恶意样本,513 个良性样本	行为特征图	ExtraTree	AUC 0.993%

本,以预测标签作为标记,使可学习的样本数量增加。样本不足限制着机器学习技术的应用,该方法为解决此问题提供了一种思路。

3 实验及分析

本节区分结构化特征及序列化特征,对 RF、LightGBM、SVM、K-means 等几种常用的机器学习方法及深度学习方法 DNN、CNN、RNN 进行分析验证。通过建立分类模型区分恶意软件与良性软件,对比不同方法的测试性能,结合实验结果分析讨论。

3.1 实验准备

2018 年,网络安全公司 Endgame 公开发布了 Ember 数据集,希望促进机器学习技术在恶意软件检测中的研究进展及应用,按照样本的出现时间,目前包括 Ember 2017 和 Ember 2018 两个部分,表 2 展示了数据集的样本数量及类别。发布者(Endgame)保证,所收集的恶意样本在 VirusTotal 上有多个检测引擎将其标记为恶意,同时良性样本在截至到收集时间为止未出现恶意标记。本节使用该数据集作为实验数据,另外收集了 2019 年间的 7 994 个恶意样本和 7 158 个良性样本,对几种主要的方法进行验证对比。

表 2 Ember 数据集情况

数据集	数量(万)		
	恶意样本	良性样本	未知样本
Ember 2017	40	40	30
Ember 2018	40	40	20

本文实验环境如下:硬件环境为 Windows10 系统,i7-8565U CPU@1.8 GHz、RAM 16.0 GB;软件环境为 Python 3.5,sklearn 0.20.3 以及其他工具包。

3.2 方法验证

过滤掉数据集中的未知样本,使用恶意样本(标记为 1)和良性样本(标记为 0)作为训练数据。根据提取特征角度的不同,本文将现有方法中使用的特征分为结构化特征和序列化特征两种类型。结构化特征是指从样本中直接提取的、具有维度意义的特征,包括解析特征、统计特征等。序列化特征是指根据样本的字节流、DLL 序列以及反汇编指令序列,间接提取的 n-gram 特征、语义特征[24]等。

首先对结构化特征相关方法进行试验验证。按照 ANDERSON H S 等[13]的方法提取了样本的解析特征和二进制字节特征。在结构化特征的应用中,树模型的性能通常优于其他类型算法,因此下一小节着重分析 RF 和 LightGBM 模型的泛化能力,另外与 SVM、K-means 和 CNN 算法进行对比。

(1)RF

按照 3:1 的比例将数据集划分为训练集和测试集,使用对数损失(Binary-log loss)作为损失函数训练随机森林模型。训练完成的模型分别命名为“rf2017”和“rf2018”。

(2)LightGBM

按照 3:1 的比例将数据集划分为训练集和测试集,使用对数损失(Binary-log loss)作为损失函数训练梯度提升决策树模型。经过参数优化,训练得到的模型分别命名为“lgbm2017”和“lgbm2018”。

其次,对序列化特征相关方法进行实验验证。本文提取 PE 文件汇编指令的 n-gram(n=3)特征,分别应用前馈神经网络(DNN)、递归神经网络(RNN)以及卷积神经网络(CNN)三种深度学习算法。Ember 数据集未提供原始的 PE 样本,仅包含特征数据,不能反汇编 PE 文件获得指令集,因此使用自行收集的 2019 年间的样本集作为实验数据,包含 7 994 个恶意样本和 7 158 个良性样本。

(1)DNN

按照 4:1 的比例将数据集划分为训练集和测试集,构建具有 5 个前馈全连接层的网络结构,训练分类模型。

(2)RNN

按照 4:1 的比例将数据集划分为训练集和测试集,构建具有 2 个递归层、1 个全连接层的网络结构,训练分类模型。

(3)CNN

按照 4:1 的比例将数据集划分为训练集和测试集,构建具有 2 个卷积层、2 个最大池化层以及 1 个全连接层的网络结构,训练分类模型。

3.3 分析对比

使用 ROC 曲线、AUC(Area Under the Curve)评价模型对阈值的敏感性和稳定性,在阈值确定的情况下,使用准确率(Accuracy)、精确率(Precision)、召回率(Recall)以及 F1-值(F1_score)评价模型的性能。用 TP、FP、TN、FN 分别表示真正例、假正例、真负例、假负例的数量,各项指标的计算如式(1)~式(4)。

$$\text{Accuracy}=\frac{\text{TP}+\text{FP}}{\text{TP}+\text{FP}+\text{TN}+\text{FN}} \tag{1}$$

$$\text{Precision}=\frac{\text{TP}}{\text{TP}+\text{FP}} \tag{2}$$

$$\text{Recall}=\frac{\text{TP}}{\text{TP}+\text{FN}} \tag{3}$$

$$\text{F1\_score}=\frac{2\times\text{Precision}\times\text{Recall}}{\text{Precision}+\text{Recall}} \tag{4}$$

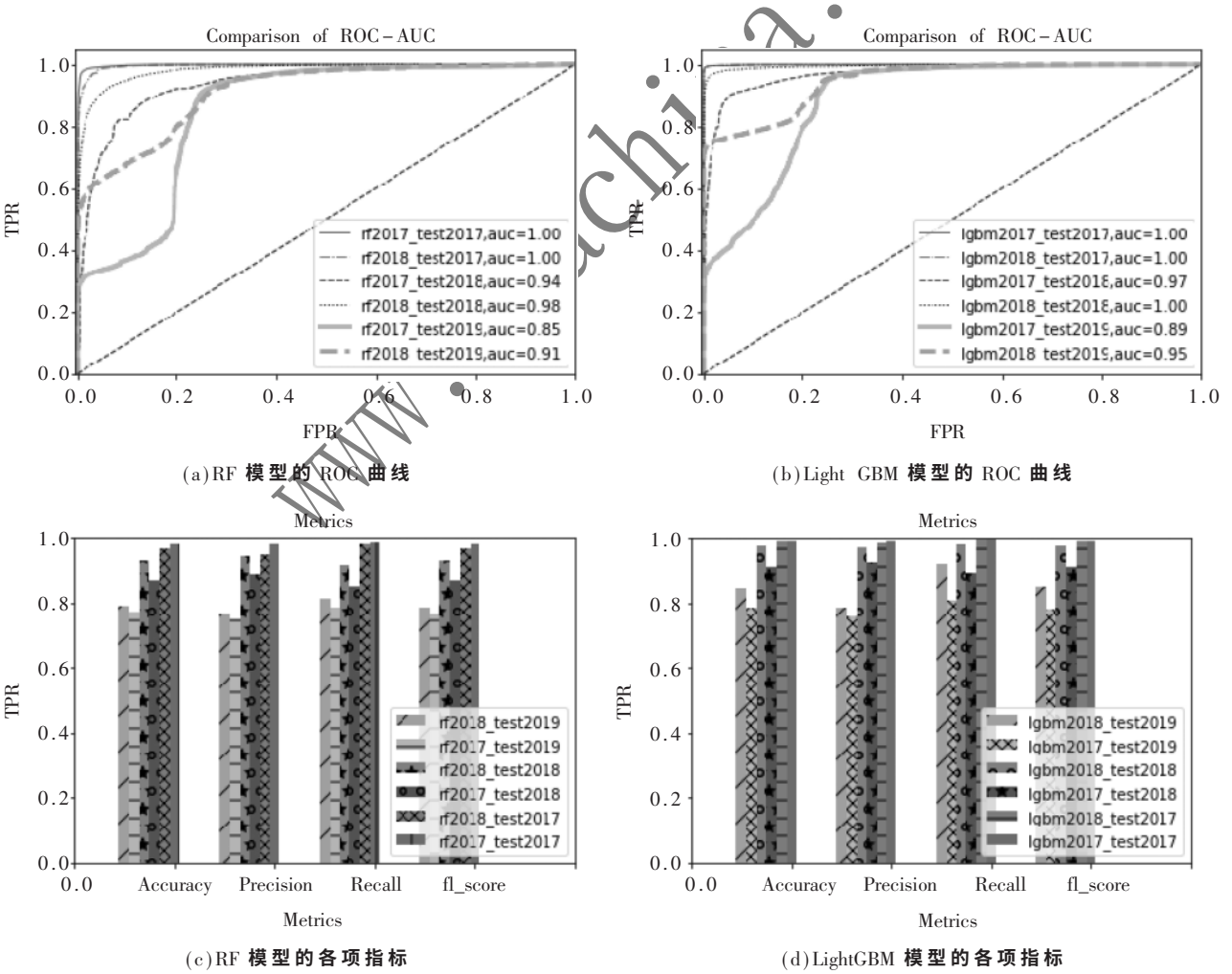
将四个使用结构化特征的模型在Ember 2017 测试集、Ember 2018 测试集以及 2019 年样本集上进行交叉检验。图 3 展示了性能评估的结果,在阈值取 0.5 时计算 Accuracy、Precision、Recall 以及 F1_score,其中,图(a)和图(c)为“rf2017”和“rf2018”的测试结果,图(b)和图(d)为“lgbm2017”和“lgbm2018”的测试结果。根据整体对比,本文得出如下结论:

(1)所有模型在 Ember 2017 测试集上的性能均优于其在 Ember 2018 测试集上的性能。据发布者(Endgame)声明,在收集 2018 年恶意软件样本时,有目的地增加了检测难度,因此在本实验中产生了这一对比结果,也反映出基于机器学习的检测方法仍然会受到恶意软件本身检测难度的影响。

(2)对比 Accuracy、Precision、Recall 以及 F1_score,可以看出,在 Ember 2017 测试集上,“rf2017”和“lgbm2017”性能优于“rf2018”和“lgbm2018”;在 Ember 2018 测试集上,结果反之。这一结果说明训练数据的覆盖情况会对模型性能产生影响,模型的检测能力依赖于对训练数据的学习,使训练数据尽可能全面地涵盖不同类型的样本,应能够增强模型的泛化能力。

(3)对比图(a)和图(b),两种算法在 Ember 2017 测试集上 AUC 均达到 1,在 Ember 2018 测试集上 LightGBM 的 AUC 略高于 RF,图(c)和图(d)中其它指标的对比也表明 LightGBM 具有更好的性能。

(4)所有模型在 2019 年样本集上平均检测准确率为 80%左右,与 2017 和 2018 年测试集上的准确



(a)RF 模型的 ROC 曲线　(b)Light GBM 模型的 ROC 曲线

(c)RF 模型的各项指标　(d)LightGBM 模型的各项指标

图 3　RF 和 LightGBM 模型的性能评估

率相比降低了约 20%,说明检测模型不足以应对样本(包括恶意样本和良性样本)的演变,例如恶意功能增强、对抗检测等。因此模型的后期维护十分重要,训练集应定期更新,另一方面,有待进一步研究更加鲁棒的检测方法。

在 Ember 2017 数据集上实验使用了 SVM(Linear svc)、K-means 和 CNN,各类算法模型准确率的对比如图 4 所示。深度学习在计算机视觉、自然语言处理及其他领域有广泛、优良的应用,前馈神经网络、递归神经网络以及卷积神经网络等,分析处理序列化数据非常具有优势。PE 文件的二进制数据流也可以作为一种序列化数据,进而应用深度学习算法,如文献[14]直接将二进制数据形式的 PE 文件作为 CNN 模型的输入,取得了 95%的准确率,在未知数据中测试的准确率为 65%至 80%。然而,由于模型输入尺寸固定的限制,该方法只能截取 PE 文件中固定大小的二进制数据流,不能获取到完整的信息。本文在应用 CNN 模型时,间接采用结构化特征,将每个维度的特征看作灰度图中一个像素。此类方法虽然避免了输入尺寸不统一以及信息损失的问题,但限制了深度神经网络应对序列化数据的优势。实验结果也表明,在本应用中树模型的表现最佳。

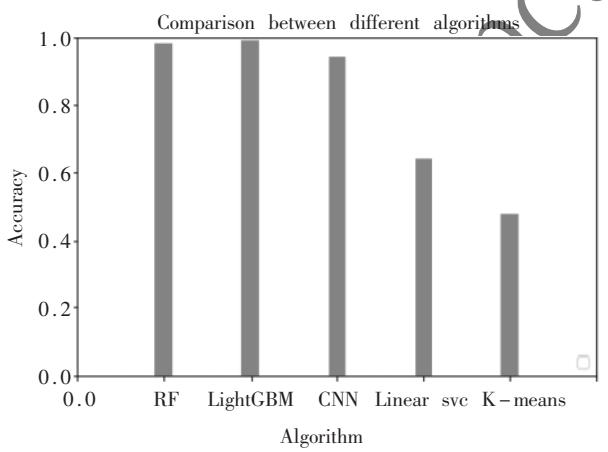
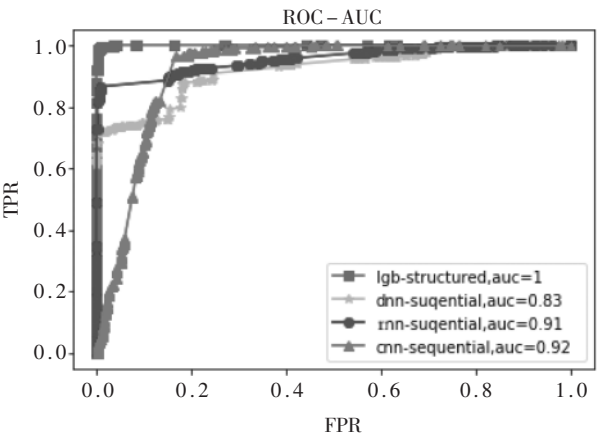
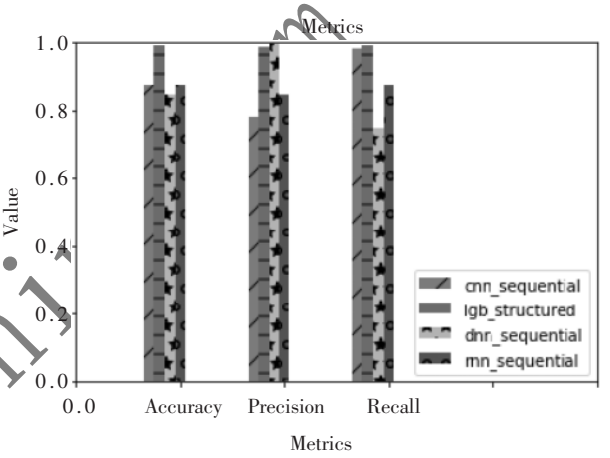


图 4 不同算法的准确性对比

使用汇编指令序列化特征的三类深度学习算法模型,在测试集上的性能评估结果如图 5 所示,同时也应用了 LightGBM 算法作为对比,本文中的 CNN 模型优于 DNN 和 RNN,AUC 值为 92%,仍然低于 LightGBM 模型。本文中的实验均为静态检测场景,未涉及动态检测方法,而动态运行中的指令序



(a)深度学习模型的 ROC 曲线



(b)深度学习模型的各项指标

图 5 深度学习模型的性能对比

列、API 调用序列应当比静态 PE 文件中的序列化特征更加具有检测意义。结合实验结果,可以得出如下结论:在静态检测中使用结构化特征,提升树模型为较优选择;深度学习方法更加适用于在动态检测场景中,使用指令运行、API 调用等序列化特征。

4 结论

本文的主要工作总结如下:(1)回顾总结了领域内的研究进展,梳理现有方法,为后续开展深入的研究提供基础;(2)对几类不同的方法做了实验验证,在不同的测试数据上交叉检验,其中,LightGBM 测试的平均准确率达到 97.06%,平均 AUC 值达到 99.25%,另外与 SVM、K-means 和深度学习算法等对比,结果表明 LightGBM 算法模型具有较好的性能。

经过调研及实验,本文指出机器学习技术应用于恶意软件检测所面临的挑战及下一步研究方向:(1)训练样本不足会影响模型的实际检测性能,由于

涉及法律法规、隐私及利益,能够获取的恶意软件样本十分有限。下一步将根据不同的类型,全面收集具有典型性的恶意软件样本,完善训练数据,以增强模型的泛化能力。(2)静态特征和动态特征的类型广泛,应用无关特征和噪声特征不仅会消耗计算资源,还会干扰模型对重要特征的学习。因此,将研究混合特征的挖掘和选择方法,使模型性能进一步提升。(3)模型结构过于复杂,参数过多,会产生一定的部署困难。研究模型结构复杂度与准确性之间的平衡,能够促进机器学习方法在本领域的应用进展。

参考文献

[1] DAMSHENAS M, DEHGHANTANHA A, MAHMOUD R. A survey on malware propagation, analysis, and detection[J]. International Journal of Cyber-Security and Digital Forensics, 2013, 2(4): 10-29.

[2] VINAYAKUMAR R, ALAZAB M, SOMAN K P, et al. Robust intelligent malware detection using deep learning[J]. IEEE Access, 2019, PP(99): 1-1.

[3] 史晓红,张艳宜.机器学习应用于恶意代码检测的研究[J].科技通报, 2013(10): 21-23.

[4] SGANDURRA D, MUNOZ-GONZALEZ L, MOHSEN R, et al. Automated dynamic analysis of ransomware: Benefits, limitations and use for detection[J]. arXiv: 1609.03020, 2016.

[5] MANGIALARDO R J, DUARTE J C. Integrating static and dynamic malware analysis using machine learning[J]. IEEE Latin America Transactions, 2015, 13(9): 3080-3087.

[6] 赵中军,曾涌泉,王运兵.基于优化 K-Means 的 Android 系统恶意软件检测的研究与设计[J].通信技术, 2018, 51(12): 212-218.

[7] 张莹.基于网络行为特征聚类分析的恶意代码检测技术研究[D].哈尔滨:哈尔滨工程大学, 2018.

[8] DU D, SUN Y, MA Y, et al. A novel approach to detect malware variants based on classified behaviors[J]. IEEE Access, 2019(7): 81770-81782.

[9] 张玉玲,尹传环.基于 SVM 的安卓恶意软件检测[J].山东大学学报(工学版), 2017, 47(1): 42-47.

[10] 翟红玉.基于 SVM 的网络恶意程序检测方法研究[J].网络安全技术与应用, 2015, 180(12): 77-78.

[11] 李涛.基于 SVM 的恶意 PDF 检测研究[J].现代计算机(专业版), 2018(8): 117-120.

[12] 杨宏宇,徐晋.基于改进随机森林算法的 Android 恶意软件检测[J].通信学报, 2017, 38(4): 8-16.

[13] ANDERSON H S, ROTH P. EMBER: an open dataset for training static PE malware machine learning models[J]. arXiv: 1804.04637, 2018.

[14] 王蕊,冯登国,杨轶,等.基于语义的恶意代码行为特征提取及检测方法[J].软件学报, 2012, 23(2): 378-393.

[15] 李鹏,王汝传,武宁.基于空间关系特征的未知恶意代码自动检测技术研究[J].计算机研究与发展, 2012, 49(5): 949-957.

[16] WATSON M R, SHIRAZI N U H, MARNERIDES A K, et al. Malware detection in cloud computing infrastructures[J]. IEEE Transactions on Dependable and Secure Computing, 2016, 13(2): 192-205.

[17] 毛蔚轩,蔡忠闽,董力.一种基于主动学习的恶意代码检测方法[J].软件学报, 2017, 28(2): 384-397.

[18] RAFF E, BARKER J, SYLVESTER J, et al. Malware detection by eating a whole exe[J]. arXiv: 1710.09435, 2017.

[19] VINAYAKUMAR R, ALAZAB M, SOMAN K P, et al. Robust intelligent malware detection using deep learning[J]. IEEE Access, 2019(7): 46717-46738.

[20] AL-DUJAILI A, HUANG A, HEMBERG E, et al. Adversarial deep learning for robust detection of binary encoded malware[C]. IEEE Symposium on Security and Privacy Workshops, 2018.

[21] WUECHNER T, CISLAK A, OCHOA M, et al. Leveraging compression-based graph mining for behavior-based malware detection[J]. IEEE Transactions on Dependable and Secure Computing, 2017, 16(1): 99-112.

[22] DAS S, LIU Y, ZHANG W, et al. Semantics-based online malware detection: towards efficient real-time protection against malware[J]. IEEE Transactions on Information Forensics and Security, 2016, 11(2): 289-302.

[23] DAI Y, LI H, QIAN Y, et al. SMASH: a malware detection method based on multi-feature ensemble learning[J]. IEEE Access, 2019(7): 112588-112597.

[24] STEVEN H, BENJAMIN C, PHILIPPE C. Asm2Vec: boosting static representation robustness for binary clone search against code obfuscation and compiler optimization[J]. IEEE Symposium on Security and

(下转第 68 页)

IEEE, 2016.
[17] 何为.深度学习在表面质量检测方面的应用[J].机械设计与制造, 2020(1): 288-292.
[18] LIU R, GU Q, WANG X, et al. Region-concolutional neural network for detecting capsule surface defect[J]. Boletin Tecnico, 2017, 55(3): 92-100.
[19] LIU Z, ZHANG C, LI C, et al. Fabric defect recognition using optimized neural networks[J]. Journal of Engineered Fibers and Fabrics, 2019(1): 1-10.
[20] 张五一, 杨扬, 林聪, 等.基于 Gabor 滤波器组和 BP 神经网络的帘子布疵点检测研究与实现[J].中原工学院学报, 2014, 25(3): 1-6.
[21] 景军锋, 党永强, 苏泽斌, 等.基于改进 SAE 网络的织物疵点检测算法[J].电子测量与仪器学报, 2017(8): 1321-1329.
[22] 景军锋, 刘娆.基于卷积神经网络的织物表面缺陷分类方法[J].测控技术, 2018, 37(9): 25-30.
[23] 吴辰斌, 王剑.全连接神经网络算法的改进与应用研究[J].电子世界, 2019(9): 108.
[24] 景军锋, 范晓婷, 李鹏飞, 等.应用深度卷积神经网络的色织物缺陷检测[J].纺织学报, 2017, 38(2): 68-74.
[25] ZHANG D, GAO G, LI C. Fabtic defect detection algorithm based on Gabor filter and low-rank decomposition[C]. Eighth International Conference on Digital Image Processing(ICDIP 2016), 2016.
[26] 张缓缓, 马金秀, 景军锋, 等.基于改进的加权中值滤波与 K-means 聚类的织物缺陷检测[J].纺织学报, 2019, 40(12): 50-56.

(收稿日期: 2020-08-18)

作者简介:

刘艳锋(19-), 男, 硕士研究生, 主要研究方向: 图像处理、深度学习。

郑云波(19-), 男, 工程师, 硕士研究生, 主要研究方向: 新型功能性纺织品。

韩军(19-), 通信作者, 男, 研究员, 主要研究方向:。E-mail: junhan@fjirsm.ac.cn。

(上接第 44 页)

Privacy(S&P). Washington, DC, USA, IEEE Computer Society, 2019: 38-55.

(收稿日期: 2020-09-01)

作者简介:

景鸿理(1962-), 男, 硕士, 高级工程师, 主要研究方向: 操作系统、信息与网络安全、密码理论及应用。

黄娜(1990-), 通信作者, 女, 博士, 主要研究方向: 机器学习、信息与网络安全。E-mail: na.huang@qq.com。

李建国(1964-), 男, 硕士, 高级工程师, 主要研究方向: 网络安全、物联网安全、应用密码学、机器学习与网络安全。

(上接第 49 页)

能驱动型网络防御设备[EB/OL]. [2019-10-31]. http://www.dsti.net/Information/News/117130.
[11] 胡影, 上官晓丽, 张宇光, 等.人工智能安全标准现状与思考[J].保密科学技术, 2017(11): 23-26.
[12] 胡金锁, 张迎, 葛玉, 等.陆军武器装备建设智能化转型思考[J].国防科技, 2019(8): 20-23.
[13] 李风雷, 卢昊, 宋闯, 等.智能化战争与无人系统技术的发展[J].无人系统技术, 2018(2): 14-23.

(收稿日期: 2020-09-10)

作者简介:

徐晨华(1984-), 男, 硕士研究生, 高级工程师, 主要研究方向: 电子信息产业标准化管理、标准化技术。

版权声明