# 一种基于威胁情报层次特征集成的挖矿恶意软件检测方法

郑 锐[1,2],汪秋云[2],林卓庞[2,3],靖蓉琦[2,3],姜政伟[2,3],傅建明[1],汪姝玮[2]
(1. 武汉大学国家网络安全学院空天信息安全与可信计算教育部重点实验室,湖北武汉 430072;
2. 中国科学院信息工程研究所,北京 100093; 3. 中国科学院大学网络空间安全学院,北京 100049)

**摘 要:** 挖矿恶意软件是近年来出现的一种新型恶意软件,其加密运算模式给受害用户带来巨大损失. 通过研究挖矿恶意软件的静态特征,本文提出一种基于威胁情报层次特征集成的挖矿恶意软件检测方法. 从挖矿恶意软件威胁情报的角度,本文分别使用字节特征层、PE(Portable Executable)结构特征层和挖矿操作执行特征层训练挖矿恶意软件分类器,利用不同恶意软件特征对恶意软件的检测偏好,使用集成方法在层次特征的基础上组建挖矿恶意软件检测器. 在实验评估中,本文使用模拟实验室环境数据集和模拟真实世界数据集进行模型性能测试. 实验结果表明,本文所设计的层次特征集成的挖矿恶意软件检测方法在模拟真实世界数据集上取得了97.01%的准确率,相对挖矿恶意软件检测基线方法获取了6.13%的准确率提升.

**关键词:** 挖矿恶意软件; 威胁情报; 机器学习; 集成学习; 深度学习; 区块链; 操作码特征
**中图分类号:** TP309.5 **文献标识码:** A **文章编号:** 0372-2112(2022)11-2707-09
**电子学报 URL:**http://www.ejournal.org.cn **DOI:**10.12263/DZXB.20211333

# Cryptojacking Malware Hunting: A Method Based on Ensemble Learning of Hierarchical Threat Intelligence Feature

ZHENG Rui[1,2], WANG Qiu-yun[2], LIN Zhuo-pang[2,3], JING Rong-qi[2,3],
JIANG Zheng-wei[2,3], FU Jian-ming[1], WANG Shu-wei[2]
(1. *Key Laboratory of Aerospace Information Security and Trusted Computing of the Ministry of Education, School of Cyber Science and Engineering, Wuhan University, Wuhan, Hubei* 430072, *China*;
2. *Institute of Information Engineering, Chinese Academy of Sciences, Beijing* 100093, *China*;
3. *School of Cyber Security, University of Chinese Academy of Sciences, Beijing* 100049, *China*)

**Abstract:** Cryptojacking malware is a new type of malware that has emerged in recent years and poses a significant threat to user host security. By studying static features of cryptojacking malware, a detection method is proposed based on integrating hierarchical threat intelligence features. We train cryptojacking malware detectors using the raw byte feature, PE (Portable Executable) parsing feature, and cryptocurrency mining operation feature, respectively. Then, the ensemble learning is used for combining these detectors to form a cryptojacking malware detector from the perspective of hierarchical threat intelligence. In the experiments, the simulated lab dataset and the simulated real-world dataset are used for performance evaluation. The experimental results show that the proposed method acquires 97.01% accuracy rate, which gets improvements of 6.13% relative to the baseline method.

**Key words:** cryptojacking malware; threat intelligence; machine learning; ensemble learning; deep learning; blockchain; opcode

## 1 引言

挖矿恶意软件是一种新型的恶意软件[1]. 挖矿恶意软件长时间大规模的计算模式会给计算机用户带来巨大的能源损失[2],某些异常的恶意软件运行逻辑甚至

收稿日期:2021-09-29;修回日期:2022-01-17;责任编辑:覃怀银
基金项目:国家自然科学基金(No.61972297, No.62172308, No.62172144); 国家重点研发计划(No.2018YFB0805005)

会给受害者造成业务损失[3]. 在静态检测方面,当前挖矿恶意软件的检测方法主要借鉴普通的恶意软件检测方法. 如使用opcode特征[4],灰度图[5]等静态特征构建分类器模型等. 但是这些方法往往忽略了挖矿恶意软件静态分析可以得到的丰富的威胁情报层次特征.

根据已有研究,挖矿恶意软件具有与普通恶意软件相似的字节码与PE(Portable Executable)结构浅层知识特征[2],同时还包含大量的与区块链计算相关的深度专家知识[6]. 根据威胁情报特征涉及的专家知识的层次不同,挖矿恶意软件特征可以分为三层:(1)二进制样本字节码. 字节码是基础的恶意软件情报特征. 字节码特征不需要专家知识,可以通过降维等方法[7,8]被转化为灰度图片特征[9],熵值特征[10]来提取其中的分类模式,同时原始字节也可以通过深度学习技术提取分类模式[11,12]. (2)PE文件结构信息. 恶意PE文件本身具有明显的情报特点[13,14],异常的段结构,文件头等信息[15]可以抽取出来作为特征向量训练识别恶意软件的分类模型. (3)挖矿操作特征. 挖矿操作所执行的区块链计算包含大量专家知识[1],需要连接加密货币基础设施来获取收益. 系统环境探测等恶意行为[16]也为挖矿操作的最终执行提供支持,这些因素反映在二进制文件静态分析上,可以转化为特定字符的匹配分析. 以上三层特征从威胁情报知识的深度来看是由浅入深的.

针对以上提到的当前挖矿恶意软件检测方法的缺陷及挖矿恶意软件本身的特点,本文从威胁情报的角度,利用三个层次挖矿恶意软件特征集成学习来构建挖矿恶意软件检测方法.

## 2 威胁情报层次特征描述

不同的静态特征训练的分类器对于恶意软件的分类具有一定的偏好. 这些静态特征可以建模不同的恶意软件特点,以达到恶意软件分类的目的. 如图1所示,按照挖矿恶意软件威胁情报提取层次划分,本文将威胁情报信息分为三层,分别是二进制文件字节码,PE结构信息,挖矿操作信息. 三层威胁情报信息分别被转换为字节特征,PE结构特征,挖矿操作特征,三层特征的解析深度及特征工程设计知识由浅入深,同时特征维度也越来越稀疏. 在特征描述维度过高时,特征提取依赖机器学习与深度学习的高维数据处理能力进行. 本节将分别介绍这几种类型特征的提取方法.

### 2.1 字节码特征层

PE文件的每一个字节均代表了具体的含义,字节特征是二进制文件的最基本的特征. 本文借助特征工程和深度学习技术获取字节码特征层次的分类模式,这些特征包括灰度图片特征,原始字节特征,以及熵值特征. 表1列出了本文所使用三种字节码特征的主要参数.

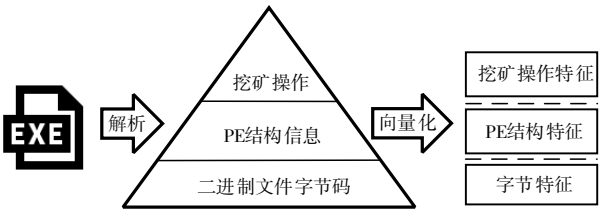


图1 挖矿恶意软件的分层特征提取

**表1 字节码特征层的特征类别**

| 特征类别 | 特征数学形式 | 特征维度 |
|---|---|---|
| 原始字节 | 一维矩阵 | 1000 kB |
| 灰度图片 | 二维矩阵 | 163 pixel×65 pixel |
| 熵直方图 | 一维/二维矩阵 | 512/16×32 |

表1中熵直方图特征可以用两种数学向量模式表示. 在本质上,字节码特征层所包含的特征主要描述了挖矿恶意软件二进制文件的相似性. 不论是图片特征还是熵值特征都旨在得到样本之间的相似性度量结果,发现其中的类别分类模式.

### 2.2 PE结构信息

PE结构也包含大量的威胁信息,某些PE信息的异常代表了恶意软件采取的攻击或逃避检测的手段. 借鉴EMBER恶意软件数据集的特征提取方法[15],PE文件的格式特征被提取作为挖矿恶意软件的分类依据. 如表2展示了这些特征的主要信息. 通过向量化与字符哈希的方式,PE结构信息可以转化为特征向量. 本文使用的PE结构信息被转化为967维的特征向量.

**表2 PE结构信息的组成**

| 特征 | 特征维度 |
|---|---|
| 一般文件信息 | 10 |
| PE头信息 | 62 |
| 节区特征 | 255 |
| 函数导出表 | 128 |
| PE文件熵特征 | 256 |
| PE文件字节统计特征 | 256 |

### 2.3 挖矿操作执行特征层

挖矿操作执行特征层描述了挖矿恶意软件4方面的特性,挖矿动作特征,恶意软件基本属性,字符规则特征,免杀特征. 这些特征被向量化作为挖矿恶意软件检测模型的训练样本.

挖矿动作特征主要涉及到软件参与区块链计算过程中所表现的具体执行特征. 加密计算具有opcode执行特性与函数调用特征,所以如果目标软件包含加密函数名称和特定opcode分布会指示被分类为挖矿恶意软件的可能性. 区块链计算需要大量的计算资源,比特

币的挖掘严重依赖GPU计算设备. 矿池设置,钱包地址等也是加密货币收益获取的重要参数,这些特征信息均可以指示待测挖矿恶意软件的嫌疑程度. 这些特征通过计数的方法转化为特征向量,挖矿动作特征一共包含24维.

恶意软件基本属性是恶意软件包含的文件信息. 这些软件基本特性能够描述恶意软件的可疑程度,资源节个数可能会指示文件释放等的恶意行为,异常的节数据也能够一定程度指示二进制文件的恶意性嫌疑. 通过归纳恶意软件的节特征,16维特征被提取出来作为分类向量.

字符规则特征提取二进制文件的特殊字符串作为特征向量,特殊字符串也是挖矿恶意软件的重要指示,挖矿恶意软件执行驻留,命令控制等操作时严重依赖文件路径,URL等字符串. 另外,挖矿恶意软件作为动作特色鲜明的恶意软件,他的字符串分布也会与其他种类样本存在差异,本文使用Yara Rule Gen工具(https://github.com/Neo23x0/yarGen)对这些特殊的字符串规则进行挖掘,生成Yara rule对待测样本进行匹配,根据匹配次数设计特征维度,这些特征维度可以有效提取高层级的字符规则语义特征. 最终使用字符规则特征方法得到了14维特征向量.

免杀特征是恶意软件的常见特征,恶意软件为了对抗调试分析和反病毒软件,常常采用检测系统环境信息来规避他们. 通过匹配恶意软件二进制文件中包含的免杀特征字符串,将匹配次数被作为挖矿恶意软件的检测特征.

表3列出了部分挖矿操作执行特征层的特征. 对于需要匹配目标字符的特征,本文通过公开渠道收集了这些字符集合. 这些收集的字符集合以及挖矿操作执行特征层提取方法已经开源(https://github.com/noideanopaper/Cryptojacking_Hunting_Ensemble_TI). 最终,对挖矿恶意软件的二进制文件抽取挖矿操作执行特征,这些特征被转换为56维向量,通过机器学习算法来训练挖矿恶意软件分类器.

**表3　挖矿操作执行特征层部分特征描述**

| 特征分组 | 特征代码 | 特征描述 |
|---|---|---|
| 挖矿动作特征 | cpu_count | "cpu"字符出现的次数 |
| | gpu_count | "gpu"字符出现的次数 |
| | opcode_var | 所有代码片段中opcode个数的方差 |
| | pool_name_count | 出现矿池地址的次数 |
| 恶意软件基本属性 | size_X | 可执行节的平均长度 |
| | rsrc_num | PE文件中资源节的个数 |
| 字符规则特征 | paths_count | 系统文件路径匹配的个数 |
| | yargen_count | 使用Yara gen工具生成的Yara规则的匹配次数 |
| 免杀特征 | av_count | Anti-virus名称字符命中的个数 |

## 3　机器学习方法及模型集成

本节首先介绍本文所涉及挖矿恶意软件特征使用的机器学习方法,然后对他们的集成学习方法进行描述,两阶段的学习任务如图2所示. 本文使用带有标签的数据训练面向识别任务的机器学习模型,所涉及的机器学习方法全部是有监督的机器学习方法.

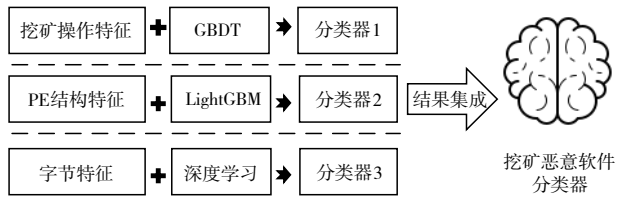


图2　机器学习方法及其集成学习

### 3.1　使用机器学习技术的任务模型

机器学习方法包括传统的浅层机器学习方法和深度学习方法. 深度学习通过增加网络层次在端到端的学习任务中将特征向量进行自适应地压缩[17],而浅层机器学习技术常常依赖于特征工程方法实现精确的特征提取,但是手工的特征缩减不可避免地会去除对分类具有作用的特征维度. 浅层机器学习技术与深度学习技术满足式(1)所规定的任务模型.

$$f_\phi(x_i^\theta)=y|_-^+ \tag{1}$$

其中,$f_\phi$是机器学习模型,$x_i^\theta$为输入的样本特征,上标$\theta$表示模型的特征类型,特征类型与训练模型存在对应关系. 下标$i$表示样本的索引. 向量$y$是机器学习输出的分类结果,输出为正例(+)或负例(-)的判别结果. 本文将挖矿恶意软件表示为正例,将非挖矿恶意软件表示为负例. 在上一节中描述的恶意软件特征具有各自的向量特点,例如灰度图片是典型的二维矩阵特征,而opcode则是序列特征. 这些特征因为各自内在的特点,所以每一种特征都具有适合自身计算特点的机器学习

模型. 表4列出了本文涉及的特征及他们使用的训练模型.

表4 不同特征对应的机器学习模型的主要结构与参数

| 特征 | 模型结构 |
|---|---|
| EH | Conv60*(2,2) + Conv200*(2,2) + Dense(500) |
| EH | Dense(1500) + Dense(1000) + Dense(500) |
| GI | 2*[Conv32(3,3)] + 2*[Conv64(3,3)] +2*[Conv128(3,3)] |
| RB | Embedding+Conv1D(128)*λConv1D(128) |
| PF | LightGBM: 100 trees, 31 leaves per tree |
| CF | GBDT: depth=4; # of leaves=20 |

如表4所示,表中列出了本文所涉及机器学习方法的模型及其参数,其中Conv代表卷积层,其后整数代表卷积核的数量,二维整数代表卷积核的大小. Dense层代表全连接层,括号中的数字代表了当前层神经元数量. 其中熵的统计直方图特征(Entropy Histogram, EH)在本文中的两种数学向量模式分别使用两种深度学习模型进行训练,灰度图特征(Grayscale Image, GI)使用卷积神经网络进行训练,字节值特征(Raw Bytes,RB)使用原始Malconv[11]进行训练. PE结构特征(PE format Features, PF)参考EMBER[15]的做法选用LightGBM(Light Gradient Boosting Machine)方法进行训练. 挖矿恶意软件操作执行特征(Cryptojacking Features, CF)中各个维度不存在关联性,所以采用梯度提升决策树(Gradient Boosting Decision Tree, GBDT)方法进行训练.

如上面的描述,在实际的方法构建中,不同的待测PE文件特征在不同的机器学习模型下进行训练,获取分类器模型. 在测试中,其所得到的单个样本的似然估计值将在下一步中输入到集成学习方法构建挖矿恶意软件检测模型.

### 3.2 模型集成

不同层次的特征对于挖矿恶意软件的描述具有不同的侧重,所以本文使用不同威胁情报层次特征训练的分类器来构建挖矿恶意软件的集成检测方法. 集成学习方法由不同层的单机器学习检测方法组合而成,数学描述如下:

$$\Pi(f_1,f_2,f_3)=y|_{-}^{+} \tag{2}$$

在式(2)中,Π表示集成学习方法,$f_\phi$表示不同特征训练的机器学习模型,三个模型输出的置信度分数组成特征向量,经过集成学习方法的运算,输出最终的正例或负例的判别结果.

在理论上所有的不同的模型都可以参与集成学习模型的计算. 但是如前所述,不同的特征层次描述了不同的恶意软件特点,这些特征分别关注到了挖矿恶意软件不同的知识层面. 所以本文采用的集成学习方法使用以威胁情报知识为层次的特征进行训练,通过使用这些特征训练的分类器输出的结果作为训练样本,之后训练集成学习模型获得最终的挖矿恶意软件分类器. 本文涉及的三层威胁情报特征共包含机器学习模型6个,三个模型组合的情况下可以得到20种组合. 根据本文提出的威胁情报分层特征组合方法得到4个组合模型,剩余16个组合为随机组合.

在集成学习方法的选择中,本文的目标是选取简单的线性模型作为集成学习方法,这样使得学习的模型更具有可解释性,以利于分类模型在现实场景中部署. 常见的线性模型方法包括支持向量机(Support Vector Machine, SVM),随机森林(Random Forest, RF),逻辑回归(Logistic Regression, LR)等. 对于支持向量机,它的输出为0或1,其输出结果不能指示恶意软件的恶意概率,只能做出定性分类. 同时支持向量机模型不具有良好的解释性[18],这对于恶意软件的这样的安全关切问题具有弱说服力,不利于模型的实际部署. 而对于逻辑回归与随机森林方法,两者都具有良好的可解释性与线性分类性能. 本文通过实验对比了两者在集成学习任务中的性能差异.

实验测试了两种机器学习分类器在不同集成学习组合情况下的性能,发现逻辑回归对于随机森林取得了明显的优势. 如表5所示,在三种模型集成的设定下,本文涉及的6种单机器学习模型可以集成为20种组合,其中4种为本文提出的三层次威胁情报特征组合. 经过测试,逻辑回归方法在全部4种三层威胁情报组合中相对随机森林获得了领先的检测性能. 在剩余的16种组合中,逻辑回归在12个组合中获得了领先的性能. 这样的性能对比与已有的研究结论一致[19]. 最终,逻辑回归被选择作为集成学习方法.

表5 随机森林(RF)与逻辑回归(LR)取得性能领先组合个数

| 组合类别 | 集成方法 | 领先组合数 |
|---|---|---|
| 威胁情报分层组合 | LR | 4 |
| | RF | 0 |
| 其他组合 | LR | 12 |
| | RF | 4 |

## 4 实验评估与分析

本节首先介绍所使用的数据集及模型性能的评价方法,之后分别在模拟实验室数据集,模拟真实世界数据集两个场景中测试本文所涉及的机器学习方法及他们的集成学习模型的性能. 同时为了说明三个层次组合的必要性,设置了对比实验来分析两层次集成方法与三层次集成方法的性能差异.

### 4.1 数据集

本文选取了两个数据集作为机器学习模型与集成学习方法的性能测试环境. 两个数据集是大数据安全

分析比赛(Datacon2020)中恶意代码分析竞赛公布的训练数据集和测试数据集. 其中训练数据集在比赛的初赛阶段公布,测试数据集在比赛后由比赛主办方公布. 这两个数据集分别记作数据集$A$和数据集$B$. 如表6所示.

**表6　Datacon2020数据集的数量分布**

| 实验 | 数据集$A$ | | 数据集$B$ | |
|---|---|---|---|---|
| | CMal | Not CMal | CMal | Not CMal |
| 原始 | 2000 | 4000 | 5898 | 11759 |
| Opcode | 1885 | 3699 | 5823 | 11120 |

由于本文的目的是构建挖矿恶意软件检测器,表中的CMal被记为正样本,Not CMal被记为负样本. 从表6中可以看出数据集$A$和数据集$B$之间的数量的比率约为1:3. 并且两个数据集中的正负样本的数量比值约为1:2. 为了模拟真实世界的挖矿恶意软件的检测任务,数据集$A$和数据集$B$分别作为模拟实验室数据集与真实世界数据集参与机器学习方法的性能测试. 现实世界的恶意软件检测器开发也遵循相同的规律,设计的方法往往在小数据集上进行训练和调优,但是需要面临在野环境下数据规模巨大的检测场景. 这样的测试范式可以有效地缓解小数据测试集可能造成的局部最优,更大程度接近真实世界中机器学习方法在挖矿恶意软件检测中的检测效果.

Datacon2020数据集的所有的标签都是由比赛组织方标定,其中非挖矿恶意样本数据集包含其他种类的恶意软件和良性软件. 比赛组织方删除了所有样本PE结构中的标志位和函数导入表. 对于需要使用分析软件对PE结构进行解析的特征类别,实验首先补全了PE,MZ两种标志位,以便反汇编软件可以执行特征的抽取. 在表6中,由于解析失败,opcode特征相对原始数据集出现了一定数量减少.

### 4.2　评价指标

恶意软件的检测实验可以采用准确率(accuracy)、精确率(precision)和召回率(recall)作为模型性能的比较标准,其中召回率指标和精确率指标是重要的性能衡量标准,前者描述检测器的检出能力,后者衡量了检测器在实际应用中的用户体验. 本文以实际应用为导向,除了在结果中对比不同方法的准确率、精确率和召回率,借鉴Datacon2020的性能指标计算方法,实验引入D-score作为第四种检测性能衡量指标. D-score主要衡量了召回率和精确率的综合评分,使用加权惩罚的方法计算挖矿恶意软件的检出性能. 计算方法如下:

$$\text{D-score}=\text{Rec}\times 100-0.9(1-\text{Pre})\times 100 \tag{3}$$

在式(3)中,Rec代表召回率,Pre是精确率,1-Pre代表误报率. Rec与1-Pre分别乘以100是为了换算成百分制,其中0.9是误报率的惩罚系数,当惩罚系数为1时表示误报的权重与检出的权重一样大,反映在实际使用场景中时,代表工具检出性能与用户体验同样重要. D-score指标相对其他检测指标能够更加直观的反映现实世界中对于检测器检测性能与用户体验的衡量.

模型的时效性也是一个分类器重要的指标. 所以实验也测试了所有方法的时效性. 时效性主要包括三种指标,即特征提取阶段时间消耗,模型训练阶段的时间消耗以及模型部署之后对样本特征向量进行计算的时间消耗. 在集成学习方法中,除了单模型所需的时间消耗之外,额外增加集成学习模型的训练时间,推理时间. 本文的实验均在Ubuntu16.04服务器平台上测试,服务器使用一个Intel i7-7700CPU,深度学习加速硬件和软件选择了NVIDIA 1080ti和CUDA 10.1. 实验代码使用Python脚本语言编写,深度学习库使用了Pytorch软件包.

### 4.3　模拟实验室数据集效果

在模拟实验室数据集上进行的实验采用交叉验证的方法来避免小数据集的随机性. 模拟实验室数据集的6000个样本被随机划分为5份,每一次测试都将其中一份作为测试数据集,其他4份作为训练数据集. 在结果展示中,取五次平均的准确率结果作为模拟实验室数据集上检测性能的比较数据. 集成学习方法的测试与单机器学习方法的测试相同,同样使用五折交叉验证的方法来测定性能指标. 对于集成学习,实验使用单机器学习模型五折交叉验证的输出作为集成学习模型的输入,这样集成学习在交叉验证的每一次测试中,4800个样本作为训练样本,1200个样本作为测试样本. 单机器学习模型性能与部分集成学习组合模型在模拟实验室数据集上的测试结果分别如表7、表8所示:

**表7　模型方法在训练数据集上执行交叉验证测试的结果**

| 特征(模型) | 准确率 | 精确率 | 召回率 | D-score |
|---|---|---|---|---|
| EH(MLP) | 0.9786 | 0.9905 | 0.9450 | 93.65 |
| EH(CNN) | 0.9770 | 0.9856 | 0.9451 | 93.23 |
| GI(CNN) | 0.9634 | 0.9802 | 0.9091 | 89.13 |
| RB(Malconv)[11] | 0.9776 | 0.9957 | 0.9371 | 93.33 |
| OP(LSTM)[4] | 0.9661 | 0.9666 | 0.9322 | 90.21 |
| PF(LGB) | 0.9873 | 0.9888 | 0.9730 | 96.29 |
| CF(GBDT) | 0.9856 | 0.9927 | 0.9640 | 95.75 |

如表7所示,表中所列出的单机器学习方法均为本文所涉及的挖矿恶意软件检测方法. 其中OP代表opcode方法,EH代表熵直方图特征方法,GI表示灰度图方法,RB表示原始字节方法[11],PF为PE结构特征,CF为挖矿操作特征. OP为本文的基线方法[4],它使用opcode序列作为特征,以双向LSTM模型作为分类模型来构建挖矿恶意软件分类器. 由表7可知,在单模型方法

上,PE结构特征与挖矿操作特征都取得了较好的效果,分别占据第一和第二. 字节码特征中的原始字节特征同样取得了较好的效果,获得了所有模型中最高的精确率,PE结构特征在D-score上和召回率上都取得了最好的性能. 在表7中,基线方法opcode特征[4]获取了较差的效果.

单模型方法组成了层次特征集成方法,三种特征层次中,字节码特征层的特征种类有四种,PE结构层特征和挖矿操作层特征都只有一种. 在实验中不同的特征层次组合被测试. 如表8所示,EHc代表CNN模型训练的熵直方图特征分类器,EHm代表使用MLP模型训练的熵直方图特征分类器. 表中列出了三种组合方法中D-score最高的前四种组合,其中CF+PF+EHm获得了最好的效果,与基线方法opcode[4]相比,组合方法获取了6.67的D-score提高,同时也好于所有的单机器学习模型的效果,特别是对于一般的恶意软件检测方法Malconv[11],其D-score分数高出3.56. 同样是三类特征集成,第五行的RB+EHm+GI组合方法检测性能较低,因为这个特征组合没有选取不同威胁情报特征层次的特征种类. 需要注意的是由于测试样本规模较小,模拟实验室数据集上的结果与模拟真实世界数据集场景下的结果并不完全相同.

**表8 部分层次集成组合模型在模拟实验室数据集的测试结果**

| 特征组合 | 准确率 | 精确率 | 召回率 | D-score |
|---|---|---|---|---|
| CF+PF+EHm | 0.9895 | 0.9954 | 0.9730 | 96.89 |
| CF+PF+RB | 0.9893 | 0.9944 | 0.9735 | 96.85 |
| CF+PF+GI | 0.9892 | 0.9939 | 0.9735 | 96.80 |
| CF+PF+EHc | 0.9892 | 0.9949 | 0.9725 | 96.79 |
| RB+EHm+GI | 0.9815 | 0.9933 | 0.9510 | 94.49 |

### 4.4 模拟真实世界数据集效果

模拟真实世界数据集的测试环境更接近机器学习模型部署之后处理的数据情况. 为了与模拟实验室数据集的效果具有对比性,模拟真实世界数据集上的测试采用相同的模型和结果处理方法. 与前述实验相同,交叉验证中的五个模型分别被作为分类模型对模拟真实世界数据集进行测试,并对获得的性能指标取平均作为模型评价的指标. 这样可以保证在统一尺度下对比模拟实验室数据集和模拟真实世界数据集上的机器学习模型的实验效果,所有单机器学习模型在模拟真实世界数据集上的检测性能如表9所示.

表9、表7相比,模拟真实世界数据集上机器学习模型出现了明显的性能下降,但是挖矿操作特征与PE结构特征依然保持了领先的性能指标. 这说明本文使用的特征层次能够明显地描述挖矿恶意软件的数据规律. 在四种字节码特征上,机器学习模型也取得了较好的成绩. 但是相对模拟实验室数据集出现了明显的性能下滑. 这可能是由于字节特征对于大规模的挖矿恶意软件数据集泛化性能不够. 使用集成学习方法可以更好地构建挖矿恶意软件的检测方法,表10展示了三个层次特征集成学习模型在模拟真实世界数据集上的测试结果.

**表9 单机器学习模型方法在模拟真实环境数据集上测试的结果**

| 特征(模型) | 准确率 | 精确率 | 召回率 | D-score |
|---|---|---|---|---|
| EH(MLP) | 0.9446 | 0.9580 | 0.8730 | 83.52 |
| EH(CNN) | 0.9436 | 0.9549 | 0.8726 | 83.20 |
| GI(CNN) | 0.9216 | 0.9539 | 0.8044 | 76.29 |
| RB(Malconv)[11] | 0.9464 | 0.9718 | 0.8651 | 83.97 |
| OP(LSTM)[4] | 0.9088 | 0.9319 | 0.7920 | 73.07 |
| PF(LGB) | 0.9641 | 0.9629 | **0.9284** | 89.50 |
| CF(GBDT) | **0.9673** | **0.9796** | 0.9215 | **90.32** |

**表10 层次特征集成方法在模拟真实世界数据集上的检测效果**

| 特征组合 | 准确率 | 精确率 | 召回率 | D-score |
|---|---|---|---|---|
| **CF+PF+RB** | 0.9701 | 0.9761 | 0.9334 | 91.19 |
| **CF+PF+GI** | 0.9700 | 0.9755 | 0.9335 | 91.15 |
| **CF+PF+EHc** | 0.9694 | 0.9750 | 0.9322 | 90.97 |
| **CF+PF+EHm** | 0.9690 | 0.9714 | 0.9346 | 90.89 |
| CF+EHc+EHm | 0.9647 | 0.9706 | 0.9223 | 89.58 |
| PF+EHm+GI | 0.9635 | 0.9615 | 0.9279 | 89.33 |
| EHc+EHm+GI | 0.9457 | 0.9517 | 0.8823 | 83.88 |

在表10中,列出了不同威胁情报层次特征的集成学习效果. 其中加粗的字体是三种特征集成中的前四名方法,可以看出这四种组合即三种威胁情报特征层次集成的4个组合,这样的结果排名与模拟实验室数据集上的集成学习测试结果相似. 在四种最优组合中,选取原始字节方法,灰度图方法,熵直方图特征的组合的检测效果依次递减. 这样的性能排序与表9中单模型在模拟真实世界数据集上的检测性能排序不相同,说明GI特征具有更高的特征集成性能,这从侧面说明,单纯堆砌最优的测试结果,在集成学习中不一定能够获得更好的效果,集成的重点还在于特征组合的种类选择. 除了列出了前四种性能最好的特征组合,表10还列出了作为对比的三种特征组合,可以看出具有相同特征层次的组合性能下降明显. 多个字节码特征的组合性能要弱于三个不同层次的组合集成方法. 相对于单模型最好的90.32的成绩,集成学习方法达到了91.19的D-score,同时相对opcode基线方法[4],本文所提出的集成学习方法获得了更优的性能,最多领先6.13%的准确率和18.12的D-score.

由表10可知相同的原始字节种类特征组成的集成方法性能与不同特征之间的集成相比较差,可以看到CF+ EHc+ EHm,PF+EHm+GI的组合效果甚至分别低于CF与PF单一模型的效果,这充分说明集成学习方法

中,单特征种类应该选择合理,否则可能会对分类性能产生副作用. 以上的性能表现验证了本节在性能评估之初的假设,拥有不同威胁情报特征层次的挖矿恶意软件集成检测方法可以获得更好的检测效果. 三种特征层次的组合对于单机器学习模型具有明显的性能提升,同时某些特征对于三层次的特征组合提升不大. 为了证明这些特征的必要性,在表11中列出了两机器学习模型组合在模拟真实世界数据集上的测试结果.

**表11　两类特征层次组合在模拟真实世界数据集上的检测效果**

| 集成方法 | Acc | Pre | Recall | D-score |
|---|---|---|---|---|
| CF+PF | 0.9700 | 0.9727 | 0.9364 | 91.18 |
| PF+RB | 0.9660 | 0.9685 | 0.9283 | 89.99 |
| CF+RB | 0.9685 | 0.9773 | 0.9273 | 90.68 |
| EHm+EHc | 0.9433 | 0.9393 | 0.8874 | 83.28 |
| EHm+GI | 0.9427 | 0.9352 | 0.8903 | 83.19 |

表11列出了部分两类特征的集成学习模型性能,其中CF+PF是最好的三类层次组合中共有的两类组合. 对于表10中性能领先的CF+PF+RB组合,D-score分数优于CF+PF组合,但是加入熵直方图特征和灰度图特征训练之后,形成的三层次组合的分类器性能弱于两类组合,说明这些特征在三类组合的集成学习模型中作用较小,在组合中妨碍了原有的特征组合分类器. 并且对比表9、表11可以发现,在二分类的实验中同样出现了组合特征性能弱于单机器学习模型的情况,EHm+EHc的组合弱于他们的单模型性能,EHm+GI同样出现了性能低于EHm单模型的结果. 这强化了模型集成中特征选择发挥了重要作用的结论.

综合以上的实验效果分析,基于威胁情报的特征层次组合相对于opcode特征[4]的基线挖矿恶意软件检测方法获得了大幅的性能提升,同时组合学习方法对于一般的恶意软件检测方法Malconv[11]也具有性能优势.

### 4.5　时效性分析

表12列出了本文所有机器学习模型的时间消耗,所有的时间测试都采用单线程的方式. 为了便于比较特征提取阶段的时间消耗,“特征提取”列给出了每种特征提取方法处理1000个样本所需的时间. 在模型训练阶段,所有的时间结果均是单次训练乘以训练轮次之后的结果. 在模型推理阶段,表中列出了在模拟真实世界数据集上,机器学习模型输出结果所用的时间. 从表12可以看出不同的机器学习所具有的时间效率,opcode的基线方法[4]由于使用了BiLSTM庞大的计算网络和IDA pro的反汇编解析时间的影响,其时效性远弱于其他的机器学习方法.

集成学习的训练和测试选取了三种机器学习模型作为集成对象,同时集成方法采用的浅层模型具有计算简单的特点,所以不同集成学习方法所消耗的训练和检测时间基本一致. 经过测试采用逻辑回归的集成学习模型训练时间为0.51705 s,并且在模拟真实世界数据集的测试时间为89.29 s. 可知,在实际的应用中,本文提出多层次特征集成学习方法在集成学习阶段不会消耗更多的时间,时间的主要消耗在于三种模型的计算时间. 以表10中的CF+PF+GI组合方法为例,总的时间消耗仍然远远优于opcode基线方法[4],并且相对原始字节方法[11]仍然具有优势(1448 s VS. 2467 s).

**表12　机器学习模型时间消耗对比**　　单位:s

| 模型 | 特征提取 | 模型训练 | 模型推理 |
|---|---|---|---|
| EH(MLP) | 39 | 117 | 17.95 |
| EH(CNN) | 39 | 193.8 | 19.35 |
| GI(CNN) | 8 | 89.64 | 17.48 |
| RB(Malconv)[11] | 0 | 1435 | 1032 |
| OP(LSTM)[4] | 3740 | 10016 | 4906 |
| PF(LGB) | 118 | 11.94 | 82 |
| CF(GBDT) | 630 | 2.40 | 63.41 |

根据本章节的检测性能与时效性能评估分析,本文所提出的威胁情报层次特征集成方法在挖矿恶意软件的检测准确率性能和时效性能上相对基线方法均取得了领先的效果. 但是这些方法所使用的恶意软件静态分析特征以及机器学习模型仍然面临一些传统的挑战,例如恶意软件的加壳加密,机器学习模型的老化问题等. 恶意软件加壳加密会对恶意软件的静态分析带来巨大的困难,导致本文提出的字符类型的静态特征难以提取. 但是根据已有的研究,现实场景下使用强加密技术(对所有的节进行加密,使得字符特征无法提取)的加壳方法较少[20],并且根据挖矿恶意软件生态调查,加壳的恶意软件只占已有挖矿恶意软件的30%[2],两个因素叠加使得现实中壳技术对本文采用的静态分析手段的影响有限. 另一方面本文训练得到的机器学习分类模型也面临着模型老化[21],对抗攻击[22]等问题,这些问题可能会导致机器学习模型性能不断下降. 对于集成学习技术来说,虽然模型老化带来了一定的性能损失,但是集成学习模型以模型的性能组合为基础,这个原理特点决定了它相对单模型方法具有更好的平衡性和鲁棒性,所以模型老化和机器学习对抗攻击对集成学习的影响也相对单机器学习模型更小.

## 5　结论

当前挖矿恶意软件检测主要借鉴普通恶意软件检测的特征设计方法,这样的特征设计方法容易导致真实环境下的模型性能大幅衰减. 本文利用挖矿恶意软件的威胁情报特性. 通过字节码特征层,PE结构特征层,挖矿操作执行特征层的集成学习设计了挖矿恶意

软件检测方法,实验结果表明,所提出的方法在模拟真实世界的数据集上的表现大幅优于基线方法.本文所提出的检测方法对于现实中挖矿恶意软件检测器的构建具有一定的参考意义.

**致谢** 感谢奇安信公司开源了供科学研究的挖矿恶意软件数据集.

## 参考文献

[1] TEKINER E, ACAR A, ULUAGAC A S, et al. Sok: cryptojacking malware[C]//2021 IEEE European Symposium on Security and Privacy(EuroS&P). Vienna: IEEE, 2021: 120-139.

[2] PASTRANA S, SUAREZ-TANGIL G. A first look at the crypto-mining malware ecosystem: A decade of unrestricted wealth[C]//Proceedings of the Internet Measurement Conference(IMC). Amsterdam: ACM, 2019: 73-86.

[3] 安天.六小时处置挖矿蠕虫的内网大规模感染事件[EB/OL]. (2019-09-25)[2021-09-15]. https: //antiy.cn/research/notice&report/research_report/20190925.html.

[4] YAZDINEJAD A, HADDADPAJOUH H, DEHGHANTANHA A, et al. Cryptocurrency malware hunting: A deep recurrent neural network approach[J]. Applied Soft Computing, 2020, 96: 106630.

[5] NASEEM F, ARIS A, BABUN L, et al. MINOS: a lightweight real-time cryptojacking detection system[C]//Proceedings of the 28th Network and Distributed System Security Symposium. Virtual: The Internet Society, 2021: 21-25.

[6] KONOTH R K, WEGBERG R VAN, MOONSAMY V, et al. Malicious cryptocurrency miners: Status and outlook [EB/OL]. (2019-01-29) [2021-09-15]. https://arxiv.org/pdf/1901.10794.

[7] KOLTER J Z, MALOOF M A. Learning to detect and classify malicious executables in the wild[J]. Journal of Machine Learning Research, 2006, 7(12): 2721-2744.

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

[9] KIM J Y, BU S J, CHO S B. Zero-day malware detection using transferred generative adversarial networks based on deep autoencoders[J]. Information Sciences, 2018, 460: 83-102.

[10] SAXE J, BERLIN K. Deep neural network based malware detection using two dimensional binary program features[C]//2015 10th International Conference on Malicious and Unwanted Software(MALWARE). Fajardo: IEEE, 2015: 11-20.

[11] RAFF E, BARKER J, SYLVESTER J, et al. Malware detection by eating a whole exe[C]//Workshops at the Thirty-Second AAAI Conference on Artificial Intelligence. New Orleans: AAAI Press, 2018: 268-276.

[12] RAFF E, FLESHMAN W, ZAK R, et al. Classifying sequences of extreme length with constant memory applied to malware detection[C]//Proceedings of the AAAI Conference on Artificial Intelligence. Menlo Park: AAAI Press, 2021: 9386-9394.

[13] SCHULTZ M G, ESKIN E, ZADOK F, et al. Data mining methods for detection of new malicious executables [C]//Proceedings 2001 IEEE Symposium on Security and Privacy(S&P). Oakland: IEEE, 2000: 38-49.

[14] SHAFIQ M Z, TABISH S M, MIRZA F, et al. Pe-miner: mining structural information to detect malicious executables in realtime[C]//Recent Advances in Intrusion Detection 12th International Symposium(RAID). Saint-Malo: Springer, 2009: 121-141.

[15] ANDERSON H S, ROTH P. Ember: an open dataset for training static pe malware machine learning models [EB/OL].(2018-04-16)[2021-09-15]. https://arxiv.org/pdf/1804. 04637.

[16] Microsoft Threat Intelligence Center. Threat actor leverages coin miner techniques to stay under the radar-here's how to spot them[EB/OL]. (2020-11-30) [2021-09-20]. https://www.microsoft.com/security/blog/2020/11/30/t.

[17] CHAN K H R, YU Y, YOU C, et al. ReduNet: a white-box deep network from the principle of maximizing rate reduction[EB/OL]. (2021-11-29)[2021-09-15]. https://arxiv.org/pdf/2105.10446.

[18] BELLE V VAN, CALSTER B VAN, HUFFEL S VAN, et al. Explaining support vector machines: a color based nomogram[J]. PloS ONE, 2016, 11(10): e0164568.

[19] KIRASICH K, SMITH T, SADLER B. Random forest vs logistic regression: binary classification for heterogeneous datasets[J]. SMU Data Science Review, 2018, 1 (3): 9.

[20] AGHAKHANI H, GRITTI F, MECCA F, et al. When malware is packin' heat; limits of machine learning classifiers based on static analysis features[C]//27th Annual Network and Distributed System Security Symposium. San Diego: The Internet Society, 2020.

[21] JORDANEY R, SHARAD K, DASH S K, et al. Tran-

scend: detecting concept drift in malware classification models[C]//Proceedings of the 26th USENIX Security Symposium. Vancouver: USENIX Association, 2017: 625-642.

[22] DEMETRIO L, BIGGIO B, LAGORIO G, et al. Functionality-preserving black-box optimization of adversarial windows malware[J]. IEEE Transactions on Information Forensics and Security, 2021, 16: 3469-3478.

## 作者简介

**郑　锐**　男,1992年11月出生于河南省禹州市. 现在武汉大学国家网络安全学院攻读博士学位. 主要研究方向为恶意代码分析,人工智能在网络空间安全中的应用.
E-mail: zr_12f@whu.edu.cn

**汪秋云**　男,1987年7月出生于广东省茂名市,现为中国科学院信息工程研究所高级工程师. 主要从事网络攻防对抗研究,在国内外发表学术论文近20篇,获省部级科技进步二等奖1项.
E-mail: wangqiuyun@iie.ac.cn

**林卓庞**　男,1996年9月出生于广西壮族自治区,现为中国科学院信息工程研究所硕士研究生. 主要研究方向为恶意代码检测.
E-mail: linzhuopang@iie.ac.cn

**靖蓉琦**　女,1997年6月出生于山东省泰安市,现为中国科学院信息工程研究所博士研究生,主要研究方向为恶意代码检测与分析.
E-mail: jingrongqi@iie.ac.cn

**姜政伟**　男,1985年10月出生于湖南省桂东县,现为中国科学院信息工程研究所正高级工程师,研究方向为威胁情报与威胁分析.
E-mail: jiangzhengwei@iie.ac.cn

**傅建明(通讯作者)**　男,1969年9月出生于湖南省宁乡县. 现为武汉大学国家网络安全学院教授. 主要研究方向为系统安全,网络安全等.
E-mail: jmfu@whu.edu.cn

**汪姝玮**　女,1990年7月出生于江苏省徐州市,现为中国科学院信息工程研究所工程师,主要从事恶意代码检测分析研究.
E-mail: wangshuwei@iie.ac.cn