

基于字节码的以太坊智能合约分类方法

林丹¹, 林凯欣², 吴嘉婧², 郑子彬¹

(1. 中山大学软件工程学院, 广东 珠海 519082; 2. 中山大学计算机学院, 广东 广州 510006)

摘要: 近年来, 区块链技术已在金融、医疗和政务等领域得到了广泛应用和关注。然而, 由于智能合约的不易篡改性和运行环境的特殊性, 各类安全问题频繁出现。一方面是合约开发者在编写合约时出现的代码安全问题, 另一方面是以太坊出现不少高风险智能合约, 普通用户很容易被高风险合约提供的高回报所吸引, 但对合约的风险却无从知晓。然而, 关于智能合约安全的研究主要集中于代码安全方面, 对合约功能识别的研究相对较少。假如能对智能合约功能进行准确分类, 将有助于人们更好地理解智能合约的行为, 同时保障智能合约生态安全, 减少或挽回用户的损失。已有的智能合约分类方法通常依赖于对智能合约开源代码的分析, 但以太坊发布的合约仅强制要求部署字节码, 且只有极少数合约公布了其开源代码。因此, 提出了一种基于字节码的以太坊智能合约分类方法。收集以太坊智能合约字节码和对应类别标签, 然后提取操作码频率特征以及控制流图特征; 通过实验对特征重要性进行分析, 获取适合的图向量维度及最优的分类模型; 在交易所、金融、赌博、游戏和高风险5个类别的智能合约多分类任务中进行实验验证, 使用XGBoost分类器时的F1值达到0.913 8。实验结果表明所提方法能较好地完成以太坊智能合约的分类任务, 并且能够应用于现实中的智能合约类别预测。

关键词: 区块链; 智能合约; 字节码; 分类

中图分类号: TP393

文献标志码: A

DOI: 10.11959/j.issn.2096-109x.2022046

Bytecode-based approach for Ethereum smart contract classification

LIN Dan¹, LIN Kaixin², WU Jiajing², ZHENG Zibin¹

1. School of Software Engineering, Sun Yat-sen University, Zhuhai 519082, China

2. School of Computer Science and Engineering, Sun Yat-sen University, Guangzhou 510006, China

Abstract: In recent years, blockchain technology has been widely used and concerned in many fields, including

收稿日期: 2022-05-16; 修回日期: 2022-09-01

通信作者: 吴嘉婧, wujiajing@mail.sysu.edu.cn

基金项目: 国家重点研发计划(2020YFB1006005); 国家自然科学基金(61973325); 广东省自然科学基金(2021A1515011661); 广州市科技计划项目(202102020616)

Foundation Items: The National Key R&D Program of China (2020YFB1006005), The National Natural Science Foundation of China (61973325), The Natural Science Foundations of Guangdong Province (2021A1515011661), Guangzhou Basic and Applied Basic Research Project (202102020616)

引用格式: 林丹, 林凯欣, 吴嘉婧, 等. 基于字节码的以太坊智能合约分类方法[J]. 网络与信息安全学报, 2022, 8(5): 111-120.

Citation Format: LIN D, LIN K X, WU J J, et al. Bytecode-based approach for Ethereum smart contract classification[J]. Chinese Journal of Network and Information Security, 2022, 8(5): 111-120.

finance, medical care and government affairs. However, due to the immutability of smart contracts and the particularity of the operating environment, various security issues occur frequently. On the one hand, the code security problems of contract developers when writing contracts, on the other hand, there are many high-risk smart contracts in Ethereum, and ordinary users are easily attracted by the high returns provided by high-risk contracts, but they have no way to know the risks of the contracts. However, the research on smart contract security mainly focuses on code security, and there is relatively little research on the identification of contract functions. If the smart contract function can be accurately classified, it will help people better understand the behavior of smart contracts, while ensuring the ecological security of smart contracts and reducing or recovering user losses. Existing smart contract classification methods often rely on the analysis of the source code of smart contracts, but contracts released on Ethereum only mandate the deployment of bytecode, and only a very small number of contracts publish their source code. Therefore, an Ethereum smart contract classification method based on bytecode was proposed. Collect the Ethereum smart contract bytecode and the corresponding category label, and then extract the opcode frequency characteristics and control flow graph characteristics. The characteristic importance is analyzed experimentally to obtain the appropriate graph vector dimension and optimal classification model, and finally the multi-classification task of smart contract in five categories of exchange, finance, gambling, game and high risk is experimentally verified, and the $F1$ score of the XGBoost classifier reaches 0.913 8. Experimental results show that the algorithm can better complete the classification task of Ethereum smart contracts, and can be applied to the prediction of smart contract categories in reality.

Keywords: blockchain, smart contract, bytecode, classification

0 概述

作为点对点 (P2P, peer-to-peer) 网络中的一种分布式账本技术, 区块链集成了分布式存储、密码学、共识机制等技术, 成为一种新型信任体系的构建方法。基于区块链的智能合约技术, 具有公开透明、不易篡改、经济高效等特性, 因此在诸多领域得到了广泛应用, 包括但不限于金融、交易所、社交、游戏等^[1-2]。然而, 由于这一新兴技术发展时间较短且监管尚不成熟, 一些合约可能存在安全漏洞^[3]或者欺诈行为^[4]。一方面是代码本身可能存在漏洞, 容易被犯罪分子利用从中牟利; 另一方面是可能存在欺诈合约和恶意合约, 对用户的资金产生威胁。

出于对代码安全、用户安全和区块链系统安全的考虑, 本文提出了基于字节码的以太坊智能合约分类方法。该方法可以充分整合代码逻辑信息, 实现对合约的自动化分类; 并且提醒用户哪些智能合约可能存在风险, 起到促进以太坊智能合约生态健康发展的作用。

1 研究背景

区块链是在计算机网络节点之间共享的分布

式数据库, 以数字格式保存电子信息^[5]。区块链和典型数据库的一个关键区别是数据结构^[6]。区块链会以分组 (称为块) 的形式收集信息, 并且以时间顺序将块连接起来以确保数据可靠性, 这一特性在比特币等加密“货币”系统的账本数据管理中起到了关键作用。区块链从技术上保证了数据记录的真实性和安全性, 被广泛应用在加密“货币”、去中心化金融 (DeFi, decentralized finance) 应用程序、非同质化“代币” (NFT, non-fungible token) 和智能合约中。其中智能合约和区块链的结合被视为重要的技术革新, 在工业界和学术界均备受关注。

智能合约可以理解自动执行的合约^[7], 可将买卖双方之间的协议条款直接写进代码逻辑, 而代码和其中包含的协议存在于一个分布式的区块链网络中。代码能够包含“协议”, 即在满足一定条件时自动执行提前预定好的操作, 包括转账、做决策、与其他合约交互等。目前, 以太坊是支持智能合约的最大公链平台, 最受合约发布者和用户的欢迎。智能合约在非中心化应用 (DApp, decentralized applications) 上得到了充分的发展。一个或多个智能合约以及一个用户交互界面 (UI, user interface), 就构成一个基本的 DApp。开发

者在区块链平台上部署智能合约,并且在UI界面设计调用智能合约的相关接口,来实现区块链分布式应用的多种功能。

在以太坊上,智能合约通常采用编程语言Solidity编写,由专门的编译器(solc)编译成低级字节码,并由以太坊虚拟机(EVM, Ethereum virtual machine)执行。智能合约创建后,代码会被存储在合约账户中以供调用。

1.1 国内外相关研究

以太坊的账户主要分为合约账户和外部账户。针对外部账户的分类问题,如钓鱼诈骗账户识别^[8]、账户业务分类^[9]等,主要基于交易网络和账户交易信息进行挖掘^[10-11]。相比而言,目前对合约账户的分类研究还处于早期,因为合约分类不仅需要挖掘合约账户的交易数据,而且要对合约代码进行分析。由于以太坊为当前最大的支持智能合约的平台,现有的对智能合约分类的研究大多集中在以太坊上。例如,文献[12]提出使用长短期记忆(LSTM, long short-term memory)网络对智能合约源码进行特征提取,并结合交易信息进行合约分类。然而,由于以太坊仅要求公开合约字节码,并不强制要求公开源码。因此有一些对基于字节码及翻译得到的操作码来进行合约分类的研究。Shi等^[13]提取账户特征和操作码频率特征对智能合约进行分类,实验结果表明账户特征对于合约分类任务的重要性较低,而操作码频率的重要性较高。Chen等^[14]基于账户特征和操作码特征对庞氏骗局的合约进行分类。Hu等^[15]收集了超过10 000份智能合约,首先分析了交易中的4种行为模式,然后定制了14个基本特征,使用LSTM网络进行合约分类。

1.2 研究目的

对智能合约分类的研究集中在交易特征和代码特征两方面。在交易特征方面,由于并不是所有合约账户都能提供足够的交易信息,因此通常会出现冷启动的局限。例如,在欺诈合约方面,还未开始大规模欺诈的合约并没有很多的交易信息提供,因此通过交易对这类合约进行判断通常较为滞后。但是欺诈合约的代码或者字节码肯定会被创建者提前部署上链,因此从代码特征的角度

更有可能在初始阶段检测出欺诈合约。在代码特征方面,目前的研究主要集中在源码的语义和统计操作码频率特征上。然而,以太坊智能合约并不强制公开源码,仅有极少数的智能合约公布源码,因此基于源码的自然语言处理方法不具有普适性。现有基于字节码的分析方法,提取的是统计操作码频率特征,忽略了智能合约代码结构层面上的信息。因此,如果能够在分类模型中使用必须公开的字节码来添加智能合约结构信息,可以使代码数据特征更加完整。

此外,对智能合约进行分类与识别将有助于更好地理解智能合约的行为,对智能合约的代码安全、用户安全以及区块链系统等方面都有重要意义。

1) 在代码安全方面:对于有类别信息的合约,可以有针对性地设计代码安全分析的方法,能够有效提高分析结果的准确率,提升效率。

2) 在用户安全方面:现实中的智能合约平台存在为了欺骗用户而故意标错标签的合约,智能合约自动化分类能够帮助识别合约的行为特征和真实目的,从而可以辅助用户判断合约是否为诈骗合约,一定程度上保障了用户的资产安全。

3) 在区块链系统方面:区块链中存在一些恶意合约,发出大量请求导致系统拥堵,因而智能合约的分类能够为识别此类合约提供参考。

1.3 本文贡献

智能合约的有效分类和功能识别以及人们对智能合约的行为特征的理解,有助于保障智能合约的安全,减少恶意合约对用户造成的损失,因此本文针对合约的分类问题提出相应的解决方法。总的来说,本文的主要贡献可以分为以下3个部分。

1) 数据收集。本文从XBlock^[16]、DAppRader网站和DApp网站上收集了4 757个智能合约的运行时字节码和类别信息,包括交易所、金融、赌博、游戏和高风险5个类别。

2) 算法设计。基于收集的字节码数据,本文自动化地提取了智能合约的控制流图(CFG, control flow graph)特征和操作码频率特征,其中控制流图的图特征能够丰富智能合约的语义信息

和逻辑结构。

3) 实验验证。基于提取的智能合约特征, 本文使用 XGBoost (extreme gradient boosting) 等分类算法对智能合约进行分类。实验结果表明, 所提的智能合约特征抽取算法能够有效地分类智能合约, 并应用于现实中的高风险合约识别和智能合约分类。

2 系统框架

智能合约中包含的字节码信息, 可以转换为操作码, 并进一步生成控制流图。因此, 本文考虑结合图特征和操作码频率特征来表示智能合约。

基于字节码的以太坊智能合约分类方法流程如图 1 所示。首先, 输入智能合约字节码, 利用 EtherSolve 生成每个合约的控制流图, 这个过程包括字节码的解析, 具体的字节码和操作码的转换规则如表 1 所示; 接着, 得到每个合约的控制流图后, 提取控制流图的拓扑结构并输入图嵌入模型 Graph2vec, 得到合约的图特征向量, 同时通过正则表达式获取控制流图中的操作码频率特征向量; 然后, 将上述两种特征向量结合拼接, 并对特征进行预处理和标准化; 之后, 将处理好的特征输入多种分类模型中进行智能合约分类; 最后, 利用收集的标签信息来验证分类器的效果。

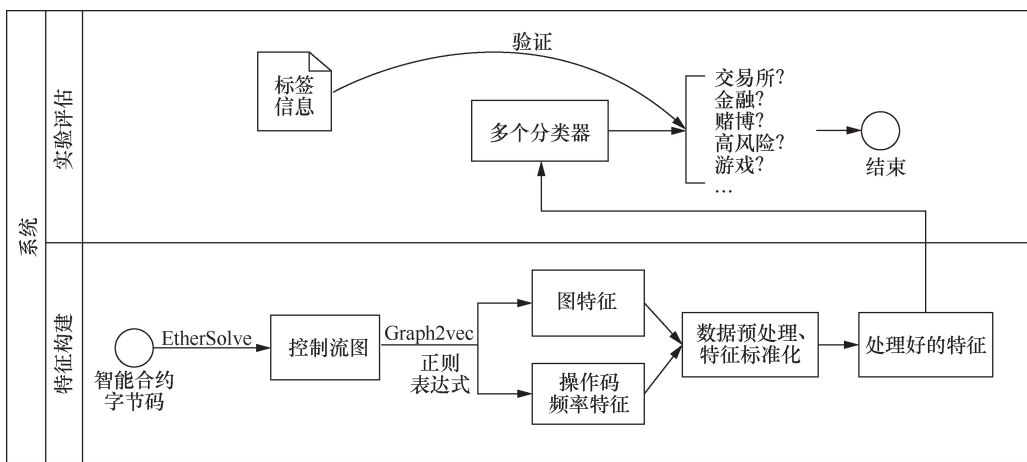


图 1 基于字节码的以太坊智能合约分类方法流程

Figure 1 Process of bytecode-based approach for Ethereum smart contract classification

表 1 字节码和操作码的转换规则

Table 1 Conversion rules for bytecode and opcode

操作码值	描述	举例
0x00~0x0B	停止和算术操作	STOP, ADD, SUB, DIV
0x10~0x1A	比较和按位逻辑操作	LT, GT, EQ, ISZERO
0x20	加密操作	SHA3
0x30~0x3E	环境信息	ADDRESS, CALLER
0x40~0x45	块信息	BLOCKHASH, COINBASE
0x50~0x5B	堆栈、内存、存储和流操作	POP, JUMP, JUMPI, JUMPDEST
0x60~0x7F	Push 操作	PUSH1~PUSH32
0x80~0x8F	复制操作	DUP1~DUP16
0x90~0x9F	交换操作	SWAP1~SWAP16
0xA0~0xA4	日志操作	LOG0~LOG4
0xF0~0xFF	系统操作	CALL, RETURN

3 算法过程

3.1 智能合约控制流图构建

关于智能合约控制流图的构建, 学术界已有不少研究和方法, 本文选择了目前较新的以太坊智能合约控制流图构建方法 EtherSolve^[17]。

本节具体介绍本文使用的基于栈符号执行的新型静态分析方法 EtherSolve, 以及其将智能合约字节码转化为控制流图的详细过程。

(1) EtherSolve

与 x86 汇编不同, 在 EVM 中没有函数的概念: 一切都是通过跳转来管理的, 既没有用于函数调用的操作码, 也没有用于从函数调用返回的操作码, 唯一可用的返回是来自外部智能合约的函数调用。这些设计决策使得对 EVM 字节码进行静态分析较为困难。

EtherSolve 是一种基于以太坊的静态分析方法, 该方法可以解析以太坊字节码中的跳转, 并为编译的智能合约构造精确的控制流图。此处的 CFG 表示执行流程的有向图, 其中的节点是程序的基本块, 边用于连接流程中可能的基本块。

(2) EtherSolve 方法流程

在 EtherSolve 中实现的静态分析由以下步骤组成。

- 1) 解析字节码: 解析字节码以识别操作码。
- 2) 构建基本块: 将操作码分组到基本块中, 按照改变程序控制流的操作码来划分基本块。
- 3) 构建基本边: 解决基本块之间基本的跳转问题, 根据每个基本块的最后一个操作码对其进行分析。
- 4) 解决孤立跳跃: CFG 构造中最具挑战性的步骤是解决孤立跳跃的目的地, 这些跳转非常常见, 如 Solidity 编译器使用函数调用后的返回值, 为了解决这部分问题, 需要进行栈符号执行。
- 5) 删除静态数据: 静态数据段与实际可执行代码分离。

6) 修饰 CFG: 对获得的 CFG 进行装饰, 以突出显示调度函数, 并确定后备功能的入口点。

使用 EtherSolve 方法后, 获取的智能合约字节码就可以转换为智能合约对应的控制流图。

3.2 智能合约控制流图嵌入

图嵌入技术可以有效地将高维稀疏图转换为低维、稠密、连续的向量空间, 最大限度地保持图的结构性质。然而, 由于需要处理的数据较多, 模型的效率是核心问题。

因此, 本文选择了可以应用于任何图任务, 且复杂度较低的方法, 即基于随机游走的 Graph2vec 方法^[18]。本节将具体介绍 Graph2vec 方法, 及其将控制流图嵌入向量的过程。

(1) Graph2vec

Graph2vec 是一种无监督的图表示学习方法, 可以学习到图的向量表示。这些图向量表示可以作为图的特征应用于后续的与图相关的任务中, 如图的聚类、分类、可视化等。Graph2vec 将自然语言处理中的词嵌入算法 doc2vec^[19]的思想迁移到图嵌入算法上, 将整个图看作一个文档, 将每个节点周围的有根子图视为文档的文字。

与图内核和其他子结构嵌入方法相比, Graph2vec 具有以下关键优势。

1) 无监督表示学习: 以完全无监督的方式学习图嵌入。

2) 与任务无关的嵌入: 由于 Graph2vec 在其表示学习过程中不利用任何特定于任务的信息(如分类标签), 因此它学习到的嵌入向量是通用的。

3) 数据驱动的嵌入: 与图内核方法不同, Graph2vec 从大量的图数据中学习图嵌入, 这使 Graph2vec 能够避免基于手工特征的嵌入方法的缺点。

4) 捕获结构等价性: 在采样时, Graph2vec 考虑了非线性子结构(如固定长度的随机游走), 确保了在学习的过程中, 结构相似的图可以产生相似的嵌入。

(2) Graph2vec 方法流程

给定控制流图的集合 $G=\{G_1, G_2, \dots\}$ 和期望输出的向量大小 δ , Graph2vec 方法的任务是为每个控制流图 $G_i \in G$ 学习一个 δ 维的实数向量表示。

Graph2vec 的基本思想是将每个节点周围的所有有根子图(即邻域)的集合视为其词汇表。随后, 通过魏斯菲尔德-雷曼内核(WL kernel, Weisfieler-Lehman kernel)^[20]生成有根子图, 并且为有根子图分配唯一的标签。接着, 在有根子图使用 doc2vec 中的 Skipgram 优化模型, 来学习数据集中每个控制流图的向量表示。

3.3 智能合约特征分类

分类方法有近年来热门的神经网络方法, 也有发展完善、基础夯实的统计方法。因此, 本文尝试了多种分类方法, 但经过实验证明, XGBoost 表现较好。XGBoost^[21]比较灵活, 而且支持自定义优化函数来实现更好的分类效果。此外, 与神经网络等方法相比, XGBoost 的速度和效率都更加优越。而与传统的决策树等方法相比, XGBoost 是多个弱分类器的结合, 分类效果会更好。

对于有 n 个样本和 m 个特征的数据集 $\mathcal{D} = \{(x_i, y_i)\} (|\mathcal{D}| = n, x_i \in \mathbb{R}^m, y_i \in \mathbb{R})$, XGBoost 可以通过计算 K 个加法函数来预测输出。

训练完成得到 K 棵树之后, 对模型输入测试样本。对于一个样本特征, XGBoost 为其在每棵

析成操作码，并提取了两类特征：一是基于 EtherSolve 方法和 Graph2vec 方法生成控制流图的图嵌入特征；二是通过正则表达式获取控制流图中的操作码频率特征。控制流图的图嵌入特征

设置为 16 维、32 维、64 维、和 128 维的向量，命名为 $x_0 \sim x_{15}$ 、 x_{31} 、 x_{63} 、 x_{127} 。操作码频率特征是 139 维的向量，均进行了数据标准化。特征提取后未标准化的数据示例如表 4 所示。

表 4 特征提取后未标准化的数据示例
Table 4 Examples of unnormalized data after feature extraction

类别	合约地址	x_0	x_{15}	...	MLOAD	SWAP2
金融	0x00000000219ab540356cbb839cbe05303d7705fa	0.44018072	0.53969365	...	158	152
交易所	0x0000000f8ef4be2b7aed6724e893c1b674b9682d	0.19015524	0.37818763	...	206	150
交易所	0x000005edbbc1f258302add96b5e20d3442e5dd89	0.27618885	0.3796904	...	214	157

4.2 评估方法

为了衡量不同检测方法对智能合约的分类效果，本文主要利用 3 个指标来对分类模型进行评价，分别是精确率 (P , Precision)、召回率 (R , Recall) 和 $F1$ 值 ($F1$ score)。

$$\text{Precision} = \frac{|TP|}{|TP| + |FP|} \quad (6)$$

$$\text{Recall} = \frac{|TP|}{|TP| + |FN|} \quad (7)$$

$$F1 \text{ score} = \frac{2 \cdot P \cdot R}{P + R} \quad (8)$$

其中，TP (true positive) 指的是真阳性，FP (false positive) 指的是假阳性，FN (false negative) 指的是假阴性，TN (true negative) 指的是真阴性。

由于本文进行的是智能合约的多分类任务，不同类别的合约在数量上存在差异，因此考虑使用指标的加权平均值对分类效果进行评估。首先计算每个类别的精确率、召回率和 $F1$ 值，再将支持度（每个类别的真实合约数）作为每个类别的权重，通过加权平均法计算分类模型最终的精确率、召回率和 $F1$ 值。

4.3 智能合约分类实验

(1) 不同分类模型的效果

本文采用多种有监督分类方法进行分类，包括决策树^[22]、随机森林^[23]、朴素贝叶斯^[24]、XGBoost^[21]、支持向量机 (SVM, support vector machine)^[25]和梯度提升决策树 (GBDT, gradient boosting decision tree)^[26]。其中每个分类模型的训练集和测试集都划分为 7:3，即测试集有 1 428 个智能合约。其中，各个模型参数如表 5 所示。

表 5 模型参数
Table 5 Model parameters

分类模型	参数
决策树	Criterion = 'entropy', max_depth = 10
随机森林	max_depth = 10
朴素贝叶斯	均为 sklearn.naive_bayes.GaussianNB 的默认值
XGBoost	learning_rate = 0.30, objective = 'multi:softprob', tree_method = 'exact'
SVM	decision_function_shape = 'ovo'
GBDT	均为 sklearn.ensemble.GradientBoostingClassifier 的默认值

不同分类模型的分类效果如表 6 所示。

表 6 各分类模型效果
Table 6 Effects of each classification model

分类模型	精确率	召回率	$F1$ 值
朴素贝叶斯 (高斯分布)	0.846 0	0.744 4	0.769 2
SVM	0.900 3	0.874 6	0.873 0
决策树	0.879 6	0.876 1	0.877 3
随机森林	0.917 4	0.894 3	0.896 0
GBDT	0.910 5	0.902 7	0.903 8
XGBoost	0.917 6	0.911 1	0.911 1

从表 6 可以看出，XGBoost 取得了最好的分类效果， $F1$ 值为 0.911 1。其他模型如随机森林、XGBoost、SVM 和 GBDT 的 $F1$ 值基本都大于 0.85，证明了本文提取特征方法的有效性。而其中效果比较好的随机森林、XGBoost、GBDT 都属于集成学习，这是因为集成学习以某种策略来组合多个模型，发挥出“集体智慧”的优势，所以在分类问题上效果通常较好。更进一步，效果最好的 XGBoost 模型在代价函数里加入了正则

项,使学习出来的模型更加简单,防止过拟合;并且借鉴了随机森林的做法,支持列抽样,不仅能降低过拟合,还能减少计算。分类效果较差的方法是朴素贝叶斯和决策树。由于朴素贝叶斯基于条件相互独立,算法较为简单,所以,效果较差。而决策树是单一的树,且限制树深度为 10,所以效果比较有限。

(2) 图向量维度的选择

通过不同分类模型的结果比较可以发现 XGBoost 是较合适的分类模型。因此本文使用 XGBoost 对不同维度的图向量,结合操作码频率特征进行分类测试。对于不同维度的图向量,智能合约分类结果如表 7 所示。

表 7 不同维度的图向量在 XGBoost 模型中的分类效果
Table 7 Effects of XGBoost classification of graph vectors of different dimensions

维度	精确率	召回率	F1 值
16 维	0.746 4	0.743 7	0.743 0
32 维	0.856 4	0.847 3	0.844 7
64 维	0.847 2	0.838 9	0.836 3
128 维	0.849 5	0.843 8	0.840 1

从表 7 可知,分类效果最好的是 32 维的图向量。但其实 32 维、64 维和 128 维的效果相差不多,原因可能是 32 维以上的向量能够充分提取控制流图的结构信息,而 16 维的特征会因为维度较小而损失一些信息。32 维的图特征能够较好地表达控制流图的结构和逻辑信息,并且数据量较小,有助于提高后面分类器的训练速度。因此,在特征重要性分析实验中采用 32 维的图向量。

(3) 特征重要性分析

通过查看特征重要性来判断,什么特征在智能合约分类上更有帮助。在 XGBoost 中排名 1~10 的特征重要性如图 2 所示,排名 48~59 的特征重要性如图 3 所示。

如图 2 所示,PUSH14、BALANCE、PUSH28、DUP7 等排名在 1~10 的操作码在智能合约分类中起到重要的作用。由图 3 可知,图的特征重要性较低。在图特征中重要性最高的为 x_35,但 x_35 在全部特征中排 48 位,重要性仅有 0.000 519 (总重要性为 1)。

图特征和操作码频率特征的重要性差别较

大,因此需要进行进一步的分析。仅使用操作码频率特征对智能合约进行分类,并选择 XGBoost 模型进行合约分类,各个类别的分类效果如表 8 所示。从表 8 可知,添加控制流图特征后的分类效果反而有所下降。

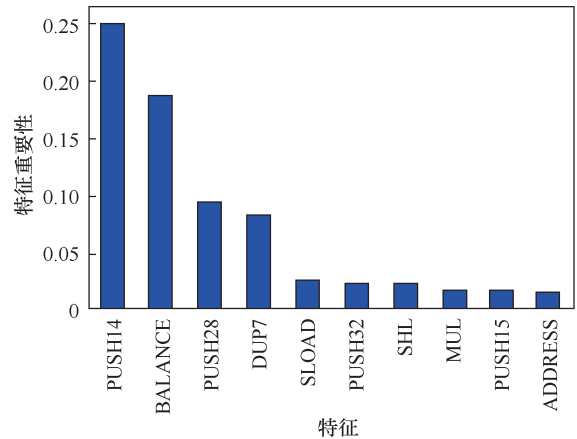


图 2 特征重要性 (排名 1~10)
Figure 2 Feature importance (Ranks 1 to 10)

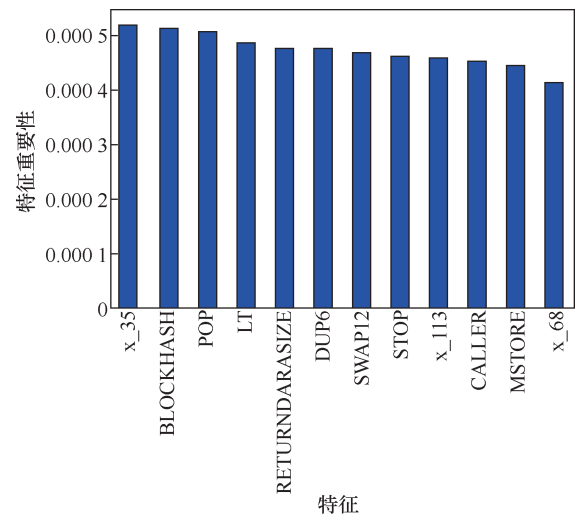


图 3 特征重要性 (排名 48~59)
Figure 3 Feature importance (Ranks 48 to 59)

表 8 仅使用操作码频率特征的 XGBoost 分类效果
Table 8 XGBoost classification effects using only opcode frequency characteristics

类别	精确率	召回率	F1 值	数据量
交易所	0.994 6	0.954 6	0.974 2	771
金融	0.705 9	0.734 7	0.720 0	49
赌博	0.937 7	0.892 0	0.914 3	287
游戏	0.785 5	0.950 4	0.860 1	262
高风险	0.787 2	0.627 1	0.698 1	59
总计	0.926 3	0.920 2	0.921 1	1 428

采用不同智能合约特征时 XGBoost 的分类结果如表 9 所示, 从表 9 可以得出 3 个结论。

表 9 采用不同智能合约特征时 XGBoost 的分类效果
Table 9 Classification effect of XGBoost using different smart contract features

采用特征	精确率	召回率	F1 值
操作码频率特征	0.926 3	0.920 2	0.921 1
控制流图特征	0.685 9	0.690 8	0.668 8
操作码频率特征+控制流图特征	0.917 6	0.911 1	0.911 1

1) 使用操作码特征来对智能合约进行分类具有较高准确率。只使用操作码特征可以较好地实现智能合约分类任务, 可见不同类别的智能合约操作码频率存在很大差异, 凭此可以有效地对智能合约进行区分。

2) 控制流图特征比操作码特征的分类效果差。仅使用控制流图特征进行智能合约分类的结果并没有比仅使用操作码特征的分类结果好。由于智能合约不同类别的表现不同, 原本预计加入能表示代码逻辑的控制流图特征, 会有良好的表现。然而, 基于控制流图特征的模型分类效果在精确率、召回率、F1 值上均比仅使用操作码特征要低。原因可能是: 许多智能合约都是实验性的, 很难从行为中检测出它们的类型; 不同类别的智能合约可能存在相似的代码, 而 Graph2vec 仅使用了节点的出入度信息和图结构信息, 这导致其提取控制流图特征的能力较为有限 (在未来的研究中有必要针对蕴含丰富节点和边特征的控制流图设计特有的图嵌入算法, 以提高智能合约分类的准确率); 当结合控制流图特征后, 分类效果有所下降。但是在只使用 32 维控制流图特征时, 分类效果在交易所、赌博和游戏类别表现都较好, 可能是因为金融和高风险类别的数据太少, 图特征分类效果较差。

3) 控制流图特征和操作码特征结合可行性。结合两个特征进行分类, 从理论上可以将代码逻辑结合到数据中, 增强模型的鲁棒性。如果能够在图嵌入模型上选择更好的方案, 或者对于数据量较少的类别获取更多数据可能会提高分类效果。

5 结束语

对智能合约进行分类与识别将有助于更好地

理解智能合约的行为, 对智能合约的安全、性能以及区块链系统等方面都有重要意义。

对智能合约自动分类与识别有助于区块链研究者和用户更好地理解智能合约的行为, 且对区块链生态上的用户安全、代码安全等方面均有重要意义。

本文收集了以太坊智能合约的标签数据集, 进而提出了一种基于字节码的以太坊智能合约分类方法。该方法基于提取的控制流图特征和操作码频率特征, 使用 XGBoost 对智能合约数据进行分类训练。实验结果表明, 该方法提取的特征比较有效, 在多个分类模型中有较为优秀的表现。而且本文所使用的字节码特征, 在任何正在运行的智能合约中都可以公开访问。同时, 通过对特征重要性的分析, 可以发现在进行智能合约分类时, 操作码频率特征比控制流图特征更为重要。然而, 仅使用操作码频率特征的分类方法可能容易遭受攻击, 攻击者可以插入无关代码进而干扰分类模型效果。通过一系列的实验验证, 本文提出的研究思路和方法有着较好的智能合约分类性能, 且适用于大部分的智能合约分类预测问题。

参考文献:

- [1] 沈鑫, 裴庆祺, 刘雪峰. 区块链技术综述[J]. 网络与信息安全学报, 2016, 2(11): 11-20.
SHEN X, PEI Q Q, LIU X F. Survey of block chain[J]. Chinese Journal of Network and Information Security, 2016, 2(11): 11-20.
- [2] 贺海武, 延安, 陈泽华. 基于区块链的智能合约技术与应用综述[J]. 计算机研究与发展, 2018, 55(11): 2452-2466.
HE H W, YAN A, CHEN Z H. Survey of smart contract technology and application based on blockchain[J]. Journal of Computer Research and Development, 2018, 55(11): 2452-2466.
- [3] 付梦琳, 吴礼发, 洪征, 等. 智能合约安全漏洞挖掘技术研究[J]. 计算机应用, 2019, 39(7): 1959-1966.
FU M L, WU L F, HONG Z, et al. Research on vulnerability mining technique for smart contracts[J]. Journal of Computer Applications, 2019, 39(7): 1959-1966.
- [4] 陈伟利, 郑子彬. 区块链数据分析: 现状、趋势与挑战[J]. 计算机研究与发展, 2018, 55(9): 1853-1870.
CHEN W L, ZHENG Z B. Blockchain data analysis: a review of status, trends and challenges[J]. Journal of Computer Research and Development, 2018, 55(9): 1853-1870.
- [5] NAKAMOTO S. Bitcoin: a peer-to-peer electronic cash system[R]. 2009.
- [6] 张亮, 刘百祥, 张如意, 等. 区块链技术综述[J]. 计算机工程, 2019, 45(5): 1-12.
ZHANG L, LIU B X, ZHANG R Y, et al. Overview of blockchain

- technology[J]. Computer Engineering, 2019, 45(5): 1-12.
- [7] BUTERIN V. A next-generation smart contract and decentralized blockchain platform[EB].
- [8] WU J J, YUAN Q, LIN D, et al. Who are the phishers? phishing scam detection on ethereum via network embedding[J]. IEEE Transactions on Systems, Man, and Cybernetics: Systems, 2022, 52(2): 1156-1166.
- [9] HUANG T, LIN D, WU J J. Ethereum account classification based on graph convolutional network[J]. IEEE Transactions on Circuits and Systems II: Express Briefs, 2022, 69(5): 2528-2532.
- [10] LIN D, WU J J, YUAN Q, et al. Modeling and understanding ethereum transaction records via a complex network approach[J]. IEEE Transactions on Circuits and Systems II: Express Briefs, 2020, 67(11): 2737-2741.
- [11] 吴嘉婧, 刘洁利, 林丹, 等. 区块链交易网络研究综述[J]. 中山大学学报(自然科学版), 2021, 60(5): 1-12.
- WU J J, LIU J L, LIN D, et al. Blockchain transaction networks: a survey[J]. Acta Scientiarum Naturalium Universitatis Sunyatseni, 2021, 60(5): 1-12.
- [12] 黄步添, 刘琦, 何钦铭, 等. 基于语义嵌入模型与交易信息的智能合约自动分类系统[J]. 自动化学报, 2017, 43(9): 1532-1543.
- HUANG B T, LIU Q, HE Q M, et al. Towards automatic smart-contract codes classification by means of word embedding model and transaction information[J]. Acta Automatica Sinica, 2017, 43(9): 1532-1543.
- [13] SHI C, XIANG Y, DOSS R, et al. A bytecode-based approach for smart contract classification[J]. arXiv preprint arXiv: 2106.15497, 2021.
- [14] CHEN W L, ZHENG Z B, CUI J H, et al. Detecting ponzi schemes on Ethereum: towards healthier blockchain technology[C]//Proceedings of the 2018 World Wide Web Conference. 2018: 1409-1418.
- [15] HU T, LIU X L, CHEN T, et al. Transaction-based classification and detection approach for Ethereum smart contract[J]. Information Processing & Management, 2021, 58(2): 102462.
- [16] ZHENG P L, ZHENG Z B, WU J J, et al. XBlock-ETH: extracting and exploring blockchain data from Ethereum[J]. IEEE Open Journal of the Computer Society, 2020, 1:95-106.
- [17] CONTRO F, CROSARA M, CECCATO M, et al. EtherSolve: computing an accurate control-flow graph from Ethereum bytecode[C]//Proceedings of 2021 IEEE/ACM 29th International Conference on Program Comprehension (ICPC). 2021: 127-137.
- [18] NARAYANAN A, CHANDRAMOHAN M, VENKATESAN R, et al. Graph2vec: learning distributed representations of graphs[J]. arXiv preprint arXiv:1707.05005, 2017.
- [19] LAU J H, BALDWIN T. An empirical evaluation of doc2vec with practical insights into document embedding generation[J]. arXiv preprint arXiv:1607.05368, 2016.
- [20] SHERVASHIDZE N, SCHWEITZER P, VAN LEEUWEN E J, et al. Weisfeiler-Lehman graph kernels[J]. Journal of Machine Learning Research, 2011, 12(9).
- [21] CHEN T, HE T, BENESTY M, et al. XGBoost: extreme gradient boosting[J]. R Package Version 0.4-2, 2015, 1(4): 1-4.
- [22] SAFAVIAN S R, LANDGREBE D. A survey of decision tree classifier methodology[J]. IEEE Transactions on Systems, Man, and Cybernetics, 1991, 21(3): 660-674.
- [23] BELGIU M, DRĂGUȚ L. Random forest in remote sensing: a review of applications and future directions[J]. ISPRS Journal of Photogrammetry and Remote Sensing, 2016, 114: 24-31.
- [24] RISH I. An empirical study of the naive bayes classifier[C]//IJCAI 2001 Workshop on Empirical Methods in Artificial Intelligence. 2001: 41-46.
- [25] SCHULDT C, LAPTEV I, CAPUTO B. Recognizing human actions: a local SVM approach[C]//Proceedings of the 17th International Conference on Pattern Recognition. 2004: 32-36.
- [26] RAO H D, SHI X Z, RODRIGUE A K, et al. Feature selection based on artificial bee colony and gradient boosting decision tree[J]. Applied Soft Computing, 2019, 74: 634-642.

作者简介



林丹 (1996-), 女, 广东揭阳人, 中山大学博士生, 主要研究方向为区块链、加密货币、网络科学的理论和应用。



林凯欣 (2000-), 女, 广东广州人, 中山大学硕士生, 主要研究方向为区块链、智能合约以及加密“货币”。



吴嘉婧 (1989-), 女, 江西吉安人, 中山大学副教授, 主要研究方向为网络科学、区块链交易网络、网络表示学习。



郑子彬 (1982-), 男, 广东潮州人, 中山大学教授, 主要研究方向为区块链、大数据、服务计算、机器学习、软件可靠性。