以太坊非法交易检测方法综述

李 梦[1], 梁广俊[2], 印 杰[2], 马 卓[2], 张 祎[3]

[1]江苏警官学院 基础课教研部 南京 中国 210031
[2]江苏警官学院 计算机信息与网络安全系 南京 中国 210031
[3]江苏省公安厅 南京 中国 210024

摘要 以太坊基于智能合约创造了一个交易生态系统, 参与者可以通过部署智能合约实现交易多元化。然而交易实体的隐蔽性为非法交易提供“便利”, 诸如传销、诈骗、蜜罐合约、洗钱、赌博和恐怖主义等违法犯罪活动频发。其中前三种是犯罪分子对正常用户单方面实施违法行为, 相较后三种而言辐射范围更广、潜在危险性更强, 故本文针对前三种非法交易行为展开研究。全文从通用检测和特殊检测两个角度对其交易特点、检测方法进行总结。首先进行通用检测研究, 通用检测关注从数据角度整理以太坊非法交易的检测方法, 发现采用监督算法(用于用户地址分类)+无监督算法(发现潜在非法用户)可实现高精度检测。然后进行特殊检测研究, 特殊检测关注特定的非法交易类型, 针对以网络钓鱼为代表的诈骗、以庞氏骗局为代表的传销和蜜罐合约交易, 分别总结其在以太坊平台上体现出的“新特点”与“新方法”。再从数据收集、特征提取、异常检测 3 个阶段综述检测技术的研究进展, 借助准确率、精度、召回率、F1-score 等评价指标进行交易类型内部和类间的比较分析, 发现在数据收集阶段采用混合采样等数据增强技术、在特征提取阶段采用图嵌入和深度学习等机器学习算法、在异常检测阶段采用集成方法等现代机器学习算法可有效提高检测精度。最后, 将视角扩展到区块链平台, 进行区块链间非法交易检测技术比较分析, 进一步给出以太坊非法交易检测未来的研究方向。

关键词 以太坊; 非法检测; 机器学习; 庞氏骗局; 蜜罐合约; 网络钓鱼诈骗
中图法分类号 TN92 DOI 号 10.19363/J.cnki.cn10-1380/tn.2024.09.10

A Survey of Ethereum Illegal Detection Methods

LI Meng[1], LIANG Guangjun[2], YIN Jie[2], MA Zhuo[2], ZHANG Yi[3]

[1] Department of Basic Course Teaching, Jiangsu Police Institute, Nanjing 210031, China
[2] Department of Computer Information and Network Security, Jiangsu Police Institute, Nanjing 210031, China
[3] Department of Jiangsu Provincial Public Security, Nanjing 210024, China

Abstract Ethereum has created a trading ecosystem based on smart contracts, where participants can diversify their transactions by deploying smart contracts. However, the concealment of trading entities provides “convenience” for illegal transactions, such as pyramid schemes, fraud, honeypot contracts, money laundering, gambling, and terrorism. Among them, the first three are unilateral illegal acts committed by criminals against normal users, which have a wider radiation range and greater potential danger compared to the latter three. Therefore, this paper focuses on the first three illegal trading acts. The paper summarizes its transaction characteristics and detection methods from two perspectives: general detection and special detection. Firstly, we conduct research on general detection, which focuses on the detection methods for sorting out illegal transactions in Ethereum from a data perspective. We found that using supervised algorithms (used for user address classification) +unsupervised algorithms (found potential illegal users) can achieve high-precision detection. Then we conduct special detection research, focusing on specific types of illegal transactions. For fraud represented by phishing, pyramid schemes represented by Ponzi schemes, and honeypot contract transactions, summarize the “new features” and “new methods” embodied on the Ethereum platform. Then, it summarizes the research progress of detection technology from three stages: data collection, feature extraction, and anomaly detection. By using evaluation indicators such as accuracy, precision, recall and F1 score, it is found that data enhancement techniques such as mixed sampling are used in the data collection stage, machine learning algorithms such as graph embedding and deep learning are used in the feature extraction stage, and modern machine learning algorithms such as integrated methods are used in the anomaly detection phase can effectively improve detection accuracy. Finally, we expand the perspective to the blockchain platform, conduct a comparative analysis of illegal transaction detection technologies between blockchains, and further provide future research directions for Ethereum illegal transaction detection.

通讯作者: 梁广俊, 博士研究生, 副教授, Email: lianggjun@126.com。

本课题得到国家自然科学基金青年基金(No. 62202209), 南京邮电大学射频集成与微组装技术国家地方联合工程实验室开放课题(No. KFJJ20200201), 江苏省教育厅科研项目(No. 2021SJA0497, No. 2023SJYB0467), 2022 年江苏高校“青蓝工程”优秀青年骨干教师项目, 江苏警官学院科研项目(No. 2020SJYZR02, No. 2023A06)资助。
收稿日期: 2022-11-16; 修改日期: 2023-03-28; 定稿日期: 2024-06-04

Key words ethereum; illegal detection; machine learning; ponzi scheme; honeypot contract; phishing fraud

1 引言

2008 年, 中本聪提出比特币[1]的概念, 用户之间可以在没有公证人的情况下进行交易, 标志着区块链的诞生。作为一种分布式数据存储技术, 区块链区别于过去以银行为代表的中心化金融交易模式, 交易过程中无需第三方公证即可产生信任, 具有去中心化、可追溯、防篡改、保证数据记录真实性和安全性等特点, 在金融、医疗保健、能源等方面均有广泛应用。作为区块链 2.0 版本, 以太坊[2]增加了智能合约功能。智能合约是通过以太坊虚拟机(Ethereum virtual machine, EVM)存储和执行的计算机程序, 可以通过交易在以太坊中达到部署、调用和删除操作, 从而实现包括金钱交易在内的交易目的, 大大拓展了业务范围。另一方面, 由于交易量的庞大与交易实体的隐蔽性, 为非法交易滋生了土壤。从全球范围看, 虚拟货币相关犯罪呈现出了多发态势。越来越多的不法分子开始利用其来进行洗钱、诈骗、赌博、恐怖主义等违法犯罪活动。2016 年的 DAO 黑客攻击[3]、2017 年的 Parity 钱包黑客攻击[4]、2022 年 UNI token 空投钓鱼等网络犯罪事件对以太坊交易环境造成了非常恶劣的影响, 总损失超 4 亿美元。根据 Chainalysis 的报告, 2021 年涉及虚拟货币的犯罪达到了创纪录的 140 亿美元[5], 以太坊的安全监管迫在眉睫。

诸多学者对以太坊的安全性研究均基于以下事实: 以太币的拥有和转移、智能合约的部署和交互等信息, 都可以在称为以太坊区块链的公共分类账上找到。这就给研究以太坊的安全性提供了素材。研究人员在以太坊区块链的安全性方面开展了多角度的研究[6-19], 主要分为两大类: 智能合约代码的安全性和以太坊交易的安全性。前者属于以以太坊为对象的犯罪行为研究, 针对智能合约代码漏洞, 研究对漏洞的检测[6-7]、针对漏洞的恶意攻击检测[8-10]和防范[11]问题, 旨在自动检测智能合约代码中的漏洞, 并在代码执行前修复合约, 不给黑客可趁之机。后者属于以以太坊为工具的犯罪行为研究, 具体表现为犯罪分子借助以太坊实施诈骗、传销、洗钱、赌博等非法行为[12-19]。针对非法交易, 学者旨在发现不法之徒的各种交易陷阱并进行事前预警和事后追踪。对于第一类已有诸多文章进行总结论述[20-26], 但是对于后者迄今尚未进行总结, 故本文将以太坊非法交易作为研究对象, 梳理基于机器学习方法进行非法交易检测的研究思路, 针对特定的非法交易类型, 对比不同的数据源、特征提取模式、异常检测方法对检测效果的影响, 并对以太坊非法交易的检测方向给出建议。

非法交易在交易数据集中属于异常数据, 如何将这些异常数据从庞大的数据集中识别出来是检测的关键问题。传统的异常检测方法有统计检验方法[27]、基于深度的方法[28]、基于距离的方法[29]、基于密度的方法[30]。这些方法通常将数据表示为特征向量, 离群点即为异常点, 在二维向量中表现良好, 但在高维向量往往难以实现精准检测, 这是由于高维向量之间存在更复杂的交互关系[31]。在以太坊交易网络中, 非法交易披着匿名的外衣伪装成正常交易, 此时它们与正常交易的特征差别不大, 无法简单依靠离群程度识别异常点[32]。这是由于后者数据集呈现极端不平衡、数据维度过高、数据量庞大的特点, 特征提取困难, 此时浅层的数据处理精准度偏低, 需要借助基于深度学习、图嵌入学习等机器学习算法挖掘深层信息, 寻找非法交易的代表性特征, 实现精准检测。

本文将分两个角度对以太坊上的非法检测问题进行总结分析。一方面是从以太坊交易数据角度出发, 观察交易过程中的行为特征, 使用通用检测方法识别状态异常的数据, 主要分为无监督学习和监督学习两种检测模型; 另一方面是针对特定的非法交易类型, 例如网络诈骗、庞氏骗局和蜜罐合约等, 根据已标记的交易数据提取相关特征, 并借助机器学习、图嵌入学习等方法进行有针对性的检测和研究。本文梳理近些年来以太坊上非法交易检测模型的研究成果, 做出如下贡献:

针对以太坊交易数据, 按照机器学习(Machine learning, ML)构建模型的思路, 结合非法检测关注异常值的特点, 分别进行数据收集、特征提取、异常检测三个阶段的方法总结, 结合模型评价尝试给出有效的检测方法。

针对以太坊上特定的非法交易, 总结其定义、特点以及对应的检测方法。第一类是以太坊上的网络诈骗, 由于诈骗是否成功主要取决于受害者对诈骗犯的信任感高低, 一旦建立信任, 以太坊上的交易只是最后一步, 难以用启发式方法寻找交易行为特征, 故诸多学者使用基于图的机器学习方法提取交易网络的高维特征, 包括图嵌入技术进行特征压缩、图神经网络进行数据表达等, 寻找具有代表性的诈骗特征。

第二类是以太坊传销。区别于诈骗的单向性, 传销具有层级关系, 通过拉人返佣金以及资金冻结的方式留住受害者。本文首先总结传销在以太坊上的新特点, 以内核——庞氏骗局为例展开研究, 其检测模型根据是否提取交易数据特征分为全特征模型(考虑交易信息和智能合约信息)和 0-day 模型(仅考虑智能合约信息)。

最后一类是蜜罐合约, 一种伪装有漏洞的陷阱合约, 具备诱导性。其研究将根据部署主体分为恶意蜜罐检测模型(蜜罐由攻击者部署, 意在吸引受害者上当)和良性蜜罐部署模型(蜜罐由学者部署, 意在收集合约攻击者信息)进行分析。

本文内容安排如下。第二节对以太坊基本概念、以太坊上非法交易以及模型评价指标进行简单介绍; 第三节从以太坊交易数据出发, 总结非法交易的通用检测方法; 第四节从特定交易类型出发, 分别讨论以太坊上网络诈骗(以网络钓鱼诈骗为代表)、传销(以庞氏骗局为代表)以及蜜罐合约的交易新特征及检测方法的总结与对比分析; 第五节和第六节分别进行以太坊内部、以太坊与其他区块链平台的非法交易检测方法对比; 第七节对以太坊非法检测模型的趋势和挑战进行阐述; 最后, 在第八节给出对本文的总结。

2 预备知识

2.1 以太坊

以太坊的概念由 Buterin [33]于 2013 年首次提出。现今它是市值仅次于比特币的第二大虚拟货币, 也是最大的支持智能合约的区块链平台。在以太坊上, 不仅可以使用以太币(ETH)进行交易, 还可以通过以太坊虚拟机(EVM)运行智能合约。如图 1 所示, 以太坊区块链是一种运行在区块链上的完整编程语言, 帮助开发者构建和发布分布式应用程序。本文主要从数据角度研究以太坊上非法交易的检测, 故重点分析数据层及交易网络。

2.2 账户与智能合约

区块链根据交易类型的不同, 可以分成两大类: 基于账户的(例如以太坊)和基于未用交易输出(Unspent Transaction Output, UTXO)的(例如比特币)。在基于账户的以太坊中, 一个交易只有一个输入和一个输出地址, 这种方式类似于账户之间的点对点交易, 地址可以重复使用, 可以用传统的社交网络研究方法对其进行研究; 而在基于 UTXO 的比特币等区块链中, 一个交易往往对应多个输入和输出地址, 且地址的二次利用率不高, 使网络分析复杂化。以太坊中基于账户的平台设计对区块链网络有深远影响。

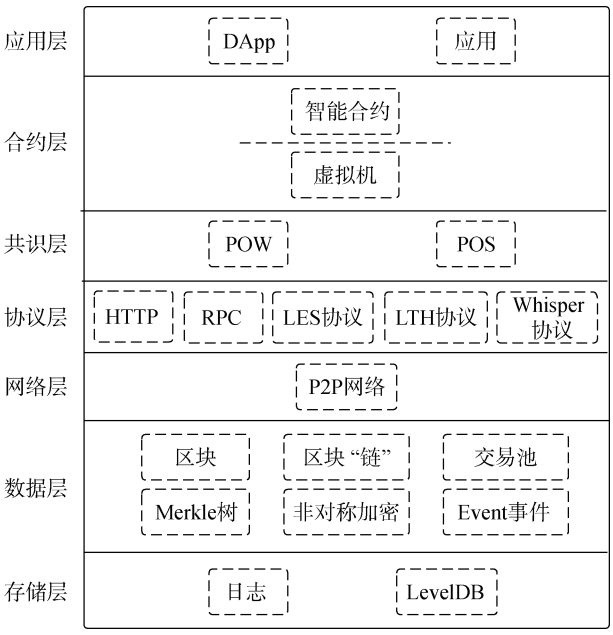

图 1 以太坊主要架构

Figure 1 Main architecture of Ethereum

2.2.1 以太坊账户

以太坊平台以账户为基本对象进行信息和价值转移[34], 主要有两类账户: 智能合约账户(Contract account, CA)和外部拥有账户(Externally owned account, EOA)。两者的主要区别在于智能合约账户由智能合约的可执行代码控制, 而外部拥有账户由持有公私密钥对的人控制。CA 由智能合约控制, 本质上是一个执行程序, 通常使用高级语言(主流语言为 Solidity)进行源代码开发, 通过 EVM 编译器将其转换成字节码, 然后通过客户端将编译好的字节码上传到以太坊。智能合约部署到以太坊后, 当触发条件满足时可以自动执行。触发条件包括外部拥有账户的调用以及其他智能合约的调用。虽然智能合约在部署后无法修改, 但以太坊允许合约自毁, 可用于停止有缺陷的合约等。EOA 即用户账户, 用户首先通过计算私钥的哈希值生成公钥, 然后用公钥的后 160 位作为 EOA 地址。地址是 EOA 的唯一标识符。虽然 EOA 没有可执行代码, 但它可用于存储当前的以太币余额或转移以太币、部署合约以及通过调用智能合约执行交易。

图 2 显示 EOA 和 CA 的状态, 包含四大元素:

已执行交易数: 表示该账户发出的交易数量。

持币数: 记录该账户拥有的以太币余额, EOA 和 CA 均可持有以太币。

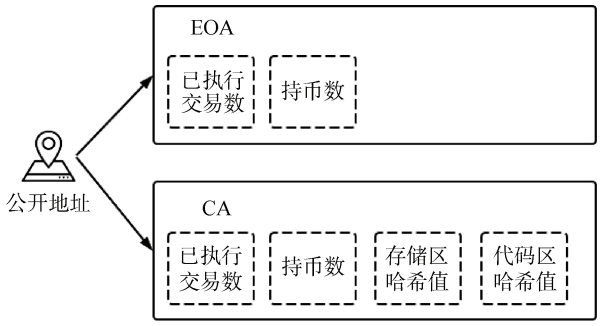

图 2　以太坊账户分类

Figure 2　Ethereum account classification

存储区哈希值: CA 独有, 智能合约运行中产生的数据存放在存储区内, 通过散列函数得出校验哈希值。

代码区哈希值: CA 独有, 代码区即为智能合约代码本身, 在合约创建周期内不会改变。代码区数据通过散列函数得出校验哈希值。

2.2.2　以太坊智能合约

智能合约可以定义为在区块链上以数字方式促进、验证和执行双方或多方之间订立的合约的计算机协议。部署成功的智能合约以 EVM 字节码的形式存在于区块链中。以太坊是目前最流行的智能合约开发平台, 可以用于设计各种去中心化应用程序, 例如数字版权管理、众筹、赌博等。

对智能合约的处理有两种方式: 调用和交易。调用是合约的本地功能, 不会对外广播, 故不会更改任何数据信息; 交易会产生以太币转移, 对应交易信息会被矿工打包上链。接下来介绍智能合约的完整生命周期。

智能合约的链上生命周期包括三个阶段: (1)智能合约的部署。智能合约创建者将编写好的智能合约代码发送到 0 地址, 经过验证后即可上链, 成为智能合约账户, 此时所有账户均可通过区块链访问合约。(2)智能合约的执行。智能合约部署后, 合同条款已被监控和评估。一旦满足合同条件, 合同程序(或功能)将自动执行, 进行的交易信息和状态更新亦随之存储在区块链上。(3)智能合约的完成。智能合约执行后, 所有相关方的新状态都会更新。因此, 智能合约执行期间的交易以及更新的状态都存储在区块链中。

目前 Solidity 是开发新型智能合约项目的主流语言, 而以太坊则是首个利用 Solidity 语言编写智能合约的平台。基于以太坊, 发展出了一系列智能合约平台, 如上表所示, 大部分支持 Solidity 语言编写。故对以太坊智能合约代码数据进行分析有助于整理 Solidity 语言逻辑, 研究结果可以迁移到其他支持 Solidity 语言开发的区块链交易平台, 有助于非法交易的检测和预警。

表 1　区块链平台的智能合约语言

Table 1　Smart contract language of blockchain platform

语言	区块链平台
Solidity	以太坊、Quorum、Wanchain、æternity、Counterparty、Rootstock、Qtum、Cardano、Soil、Expanse、Ubiq、Ethereum Classic、Monax。
RHOLang	RChain
C++, C	EOS、Stratis、Burst
GoLang, Node.js	HyperLedger Fabric
多种语言	Neo、NXT、Nem、Stellar

2.3　以太坊交易

交易在以太坊中表示从一个账户地址发送到另一个账户的信息。

2.3.1　以太坊交易

根据以太坊黄皮书定义, 交易指一段数据, 由外部拥有账户持有人签名发起。交易内包含一个消息或一个智能合约。根据发送方可将交易分为外部交易(发送方为 EOA)和内部交易(发送方为 CA), 如图 3 所示。

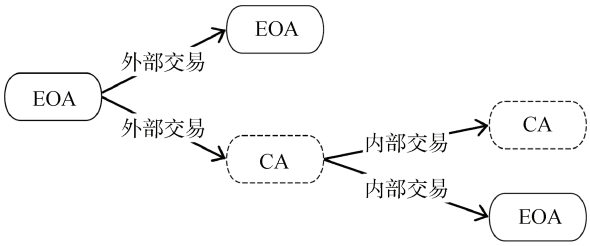

图 3　以太坊交易

Figure 3　Ethereum transaction

常见交易有三种:

以太币转账: 从一个账户向另一个账户发送以太币。例如, EOA 到 EOA。

智能合约部署: 将智能合约的数据整合到交易体的数据区, 并向 0 地址发送。一旦交易被捕获且挖矿完成, 合约就成功部署在区块链上。

智能合约调用: EOA 发送消息到已部署的 CA, 从而激活 CA 代码, 执行各种操作。

三种交易中, 前两次交易属于外部交易, 交易信息将会上链, 最后一种交易属于内部交易, 交易信息不会上链, 除非智能合约强制执行金融交易。发起一次交易包含下列信息:

None: 表示该账户发起的交易数。

GasPrice: 发起交易需要支付的交易费。一般智能合约部署花费最高。

signature: 本次交易发送方的标识符, 由发送方利用私钥签名时生成。

recipient: 本次交易的接收地址。若本次交易为以太币转账, 则接收方为账户地址; 若本次交易为智能合约创建, 则接收方地址可空缺(为 0); 若本次交易为智能合约调用, 则接收方为 CA 地址。

value: 交易中包含的以太币数值。

data: 交易中包含的数据。若本次交易为以太币转账, 该字段可空缺; 若本次交易为智能合约创建, 则该值包含编码后的函数名和参数的字节码; 若本次交易为智能合约调用, 则该值包含初始化合约的字节码。

交易自创建后, 经过广播交易、网络扩散、矿工挖矿记账、参与共识算法挑选阶段, 最终进入区块链并永久保存。

任何用户均可创建智能合约, 而智能合约的部署信息和交易信息均在区块链上永久保存和公开, 故以太坊中的交易数据存在智能合约代码数据(智能合约部署阶段生成的数据)、链上交易数据(交易阶段生成的数据)两种形式。

交易操作主要有呼叫、创建、奖励和自毁[28]等。基于交易, 可以在以太坊上进行诸如以太币转账、智能合约创建调用等操作。故研究以太坊上的交易有助于进行非法活动的特征分析、行为捕捉和实时追踪, 从而达到对以太坊交易环境的监管。

2.3.2 以太坊交易网络

Akcora 等[34]提出, 与涉及少至两个或多达数千个地址的 UTXO 交易不同, 账户区块链上的交易只涉及两个地址: 发送方和接收方。构成的交易网络有三种: (1)硬币交易网络。与 UTXO 类似, 节点为交易地址, 边缘仅包含交易信息。(2)令牌交易网络。由智能合约创建的交易网络。其中令牌表示在给定项目的生态系统内使用的数字资产。(3)跟踪网络。不同于前两种交易网络, 主要体现地址之间的功能性交互关系。以太坊上非法交易主要表现为前两种交易网络。

硬币交易网络指用以太币进行交易的网络, 由 EOA 发起, 包括 EOA 到 EOA、EOA 到 CA、EOA 到 0 地址三种端到端连接; 边缘包括交易值、账户随机数、交易费、时间戳等信息。可看出代笔交易网络属于定向加权多重图。由于以太坊不鼓励新建地址, 用户可能会长期持有同一账户地址进行长期交易; 另一方面, 智能合约一经部署就会保存在区块中, 无法人为更改。故代币交易网络数据均存在于区块中, 有助于学者及时跟踪节点行动和网络动态。

令牌交易网络指用代币进行交易的网络, 属智能合约平台特有。为了扩大以太坊的交易业务范围, 用户通过部署智能合约来创建令牌(智能合约中包含令牌的功能和业务逻辑), 例如首次代币发行。任何区块链参与者都可以创建令牌并促进其交易。令牌交易网络以 EOA、NULL 和智能合约地址作为节点。交易类型有三种: (1)令牌初始化: 通过创建智能合约生成令牌, 定义令牌属性。(2)代币交易: 用户用以太币“购买”代币, 或将代币“兑现”为以太币。(3)代币管理: 智能合约创建者对发行的代币进行处理。该交易可能会删除合约或将其余额(以太币或代币)转发到另一个地址。然而, 代币交易是一种内部交易, 不会以普通以太坊交易的形式向网络广播。换句话说, 代币交易实际上是对代币智能合约变量余额的更新。

2.3.3 以太坊交易类型

区块链的匿名化给链上交易蒙上了一层神秘的面纱, 若能实现账户识别, 有助于对以太坊的风险评估和市场监管。

Klusman 等[35]发现无法简单将比特币去匿名化技术迁移至以太坊。Victor[36]提出了与存储地址、空投多重参与和令牌授权机制相关的模式的启发式方法, 对地址进行聚类来识别以太坊实体。Shin 等[37]比较分析了比特币和以太坊的钱包地址聚类方法.Wu 等[38]和 Dyson 等[39]通过追踪交易识别所有者账户。Lin 等[40]研究资金流的可解释策略, 证明交易频率和交易量都会影响以太坊中新交易的生成。这意味着在跟踪以太坊账户之间的资金流时, 应该更加关注那些时间间隔更短、金额更大的交易路径。

Bang 等[41]和 Lee 等[42]监控区块链上生成的区块和交易, 并检测非法交易。Zhou 等[43]和 Huang 等[44]考虑从图分类的角度来识别以太坊账户, Zhou 等提出端到端的图神经网络框架 Ethident 刻画账户的行为模式; Huang 等利用图卷积网络来解决以太坊中的账户分类问题。通过研究, 以太坊上的交易类型主要有首次币发行(Initial Coin Offering, ICO)、采矿奖励、交易所、欺诈/黑客等。

ICO: 首次币发行是一种通过发行代币为区块链项目筹集资金的融资方法。ICO 项目通常预售代币以换取大量以太币, 一段时间后, 该项目将给支持者一定的投资回报。

采矿奖励: 以太坊通过工作量和挖矿机制保证区块链的有序性, 当有矿工成功将区块上“链”, 就

会产生新的以太币并被奖励给矿工。而个人的计算能力有限, 于是一部分人集合起来形成团队——采矿池, 采矿池将收到系统发放的大量采矿奖励, 并根据工作量证明(Proof of Work, PoW)共识协议将其分配给下属。

交易所: 交易所是为用户提供资产交易匹配和清算服务的平台。交易所账户通常与客户频繁交互, 以处理大量交易订单。

网络钓鱼/黑客: 钓客和黑客都从事非法欺诈活动, 通常会传播大量网站、电子邮件或链接, 其中包含病毒、木马、不需要的软件等。欺骗收款人直接汇款或提供系统特权的敏感信息。

2.3.4 以太坊非法交易

在以太坊的海量交易中, 掺杂着大量的非法活动。2021 年虚拟货币犯罪金额高达 140 亿美元, 较 2020 年的 78 亿美元增长了 79%, 创下历史新高。主要原因在于去中心化金融平台 DeFi 诈骗与黑客盗窃活动的激增。其中诈骗案件造成的损失高达 78 亿美元, 同比增长了 82%; 黑客盗窃案件损失高达 32 亿美元, 同比增长了 516%。对虚拟货币的研究和监管刻不容缓。

以太坊上的非法交易按照交易主体身份分为两类: 第一种是犯罪分子双方通过以太坊平台进行非法交易, 主要有洗钱、赌博等形式。第二种是交易一方为犯罪分子, 另一方为受害者, 犯罪分子通过某种犯罪手法骗取受害者的钱财, 典型的犯罪形式有诈骗、传销与蜜罐合约。以下将对洗钱、赌博、诈骗、传销与蜜罐合约进行简要介绍。

洗钱: 犯罪分子为了使从非法活动中获得的大量资金看起来合法, 通过买入虚拟货币, 再出售转化为法定货币实现洗白。包括三个阶段: 放置、分层和集成。在放置阶段, 脏钱被引入合法的金融体系; 再分层阶段, 犯罪分子运用分拆技术进行多次交易, 实现难以被追溯的目的; 最后, 在集成阶段, 通过进一步的交易将其集成到某一地址账号进行提取。

赌博: 当区块链与赌博结合起来, 具备新的特点: (1)场景不同, 赌博平台利用智能合约开设网络赌场, 从而获得赌客信任; (2)赌博形式不同, 平台借助“区块链+游戏”, 采取“竞猜博彩”赌博新形式, 类似“赛马”等项目吸引用户参与; (3)投注对象不同, 利用虚拟货币市场价格涨跌设置杠杆合约进行赌博; (4)筹码不同, 在区块链上的赌博一般以虚拟货币作为筹码进行支付结算。

诈骗: 犯罪分子故意欺骗他人以获取非法或不正当利益。而以太坊诈骗中, 欺诈者利用了区块链技术的独特特征, 借助智能合约等工具实施欺诈, 实现诈骗目的。

传销: 传销组织将虚拟货币作为工具或者以虚拟货币为噱头, 以投资虚拟货币高额返利为诱饵, 借助互联网平台拉人头形成层级关系, 以发展下线的数量和投资虚拟货币的价值为返利依据的一种网络传销[45]。

蜜罐合约: 犯罪分子通过部署假装赠送资金但实际上包含隐藏陷阱的合约来引诱用户进入陷阱。这种新型合约通常被称为蜜罐合约。

综上所述, 洗钱和赌博是犯罪分子的行为, 参与双方皆对该行为的违法本质有所了解, 属于双向非法交易; 而诈骗、传销和蜜罐合约是犯罪分子单方面实施违法行为, 其目标群众是以太坊上 90%以上的正常用户, 属于单向非法交易。相较而言第二种非法交易的辐射范围更广、潜在危险性更强。本文考虑影响范围和影响力度大小, 选择研究以太坊上单向非法交易(以下简称非法交易)的特点、检测方法以及未来趋势和挑战。

2.4 模型评价指标

针对非法交易的检测可视为二分类问题, 针对一次具体交易, 当该交易为非法交易时称之为阳性, 否则为阴性。故本文对模型的评价指标由四个基本部分组成: 真阳性(True Positive, TP)、假阳性(False Positive, FP)、假阴性(False Negative, FN)和真阴性(True Negative, TN), 如表 2 所示。

表 2 模型评价基本指标

Table 2 Basic indicators of model evaluation

		模型检测	
		阳性	阴性
真实结果	阳性	真阳性	假阴性
	阴性	假阳性	真阴性

假设 TP、FP、FN、TN 分别表示真阳性、假阳性、假阴性和真阴性的样本点个数, 评价模型检测效果的指标有:

准确率: $\text{Accuracy}=\dfrac{TP+TN}{TP+TN+FP+FN}$, 表示正确识别数据类别的比例, 即预测的准确度。

精度: $\text{Precision}=\dfrac{TP}{TP+FP}$, 表示一个类别的正确预测观察值与该类别下预测的总观察值之比。即, 预测为阳性的数据集中有多少是真正的阳性。

召回率: $\text{Recall}=\dfrac{TP}{TP+FN}$, 表示该类别的正确

预测观察值与该特定类别的总观察值之比。即数据集中阳性样本有多少被正确预测出来。

F1 分数: $\mathrm{F1\_score}=\frac{2\times \mathrm{Precison}\times \mathrm{Recall}}{\mathrm{Precison}+\mathrm{Recall}}$, 表示准确率和召回率的调和平均值。

3 以太坊非法交易的通用检测方法

以太坊上存在着不同类型的交易, 例如常规交易、合约部署交易、合约调用交易等。每种交易都有相似的交易流程, 故交易数据也会呈现相似的交易特征。若某笔交易的数据出现异常, 则有理由怀疑该笔交易为非法交易。

3.1 通用检测技术

通用检测技术不针对任何特定非法交易类型, 重点关注呈异常状态的交易数据。模型构建过程中主要使用机器学习技术, 故本节内容首先对机器学习模型进行简单介绍, 再根据机器学习的不同阶段依次进行技术梳理和总结。

机器学习一共经历数据收集、数据预处理、特征工程、ML 算法四个阶段[46]。在以太坊非法交易检测过程中展现出新的特点(如图 4 所示)。(1)数据收集阶段: 以太坊中的数据来源主要有交易数据(链上信息)、钱包数据(用户端信息)、非法数据(公开披露的非法交易)、合约数据(智能合约信息)。(2)数据预处理阶段: 从多渠道得到的数据需要进行预处理, 主要方法有数据清洗(删除重复信息、纠正存在的错误)、数据集成(整合多渠道信息)、数据变换(对数据进行平方根转换等, 使得数据满足方差分析的要求)、数据归约(精简数据)等。(3)特征工程阶段: 对处理后的数据集进行特征提取, 进一步地, 从多个特征(属性)中选择具有代表性的特征(包括来自交易数据的交易特征以及来自智能合约数据的合约特征等), 从而降低特征维度, 提高检测效率。(4)异常检测阶段: 筛选出合适的数据特征后, 利用有监督算法对数据集进行分类(分为合法交易集和非法交易集), 利用无监督算法发掘未知数据的属性(合法或非法)。

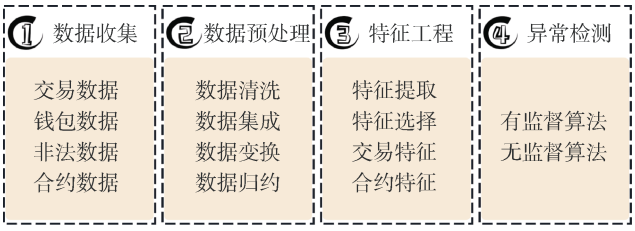

图 4 基于机器学习的通用检测流程

Figure 4 General inspection process based on machine learning

由于以太坊的外部交易数据均可在官网上查验, 且对公众开放, 故数据一般从官网 etherscan.io 即可获得; 数据预处理是考虑到数据的不完整性、不一致性, 从而对所收集数据进行分类或分组前所做的审核、筛选、排序等必要的处理, 而以太坊的数据不需要对缺失值进行转换, 且异常检测就是为了发现不同寻常的数据点, 故不会将数据预处理作为主要的研究方向, 而是专注于挖掘交易数据内部的信息, 在庞大的交易数据中寻找非法交易的踪迹。因此大部分的研究文献均着力于特征工程和 ML 算法的研究中, 以期待能够更好地挖掘出数据背后的信息, 达到检测非法交易、维护以太坊用户利益的目的。故本文沿用前人的研究方法, 梳理基于机器学习的检测模型时略去数据预处理阶段, 着重于对数据收集、特征工程、异常检测阶段的分析。

通用的检测模型从交易数据出发, 不关注具体的非法交易类型, 而是分析交易数据体现的行为模式区别, 运用机器学习等数据挖掘方法寻找异常之处(如表 3、4 所示)。O'Kane[19](2018 年)首次对以太坊上非法数据的检测开展研究, 采用 K-means 聚类算法进行无监督聚类, 检测结果欠佳, 但对以太坊上非法交易的检测就此拉开序幕。

3.2 数据收集

数据来源可分为三类: (1)交易数据[46-50]: 包括在 etherscan.io 中提取的以太坊交易、智能合约创建和调用等数据; (2)钱包数据[51]: 钱包的实例号、索引和地址等; (3)非法数据集: 包括诈骗数据集[14]、恶意地址[15,52]、恶意域名系统[53](Domain Name System, DNS)等。主要挑战包括数据量庞大、非法数据收集困难等因素。而 Morishima[17]使用 GPU 将用户图生成固定大小的子图来减轻运算负担; Sachan 等[53]将关注点转向 IP 地址, 尝试利用与账户相关联的域名中包含的信息来检测恶意分布式应用程序和恶意区块链钱包。

3.3 特征工程

特征工程包括特征提取和特征选择两个环节。学者们尝试提取包括交易特征[15,46]、局部特征[17]、结构特征[49]、时间序列特征[48,53]在内的诸多特征。主要的研究方法包括网络表示学习[46]、文本表示[50]、EM 算法[51]等。Sun 等[46]使用随机游走生成路径(Metapath2vec)学习以太坊中各个节点的特征向量, 捕获不同类型节点之间的结构相关性, 再使用主成分分析法(Principal Component Analysis, PCA)和 TSNE 算法对得到的 128 维特征向量进行降维处理。Hu 等[50]应用带有注意力层的序列学习神经网络来捕

表 3 通用检测模型建立总结

Table 3 Summary of general detection model establishment

时间	作者	数据收集	特征工程	ML 算法	检测对象
2019	Morishima[17]	--	结构特征	无监督(k-means)	交易
2019	Sun 等[46]	本地 Geth 客户端; EVM	结构特征	无监督(Birch)	用户
2019	Farrugia 等[15]	Etherscamdb; 本地 Geth 客户端	交易特征	有监督	账户
2021	Hu 等[50]	Etherscan	智能合约字节码特征	有监督	智能合约
2021	Ibrahim 等[14]	kaggle 上欺诈数据集	交易特征	有监督	交易
2021	Poursafaei 等[52]	客户端; EtherScamDB	结构+交易+邻域+本地特征	有监督	账户
2020	Poursafaei 等[47]	etherscan.io	交易+邻域+本地+时间特征	有监督+无监督	用户
2021	Agarwal 等[48,49]	Etherscan	结构+时间特征	有监督+无监督	账户
2019	Baek 等[51]	钱包	交易特征	有监督+无监督	钱包
2021	Sachan 等[53]	Etherscan; 公共来源	结构+时间特征	有监督+无监督	账户

表 4 通用检测模型评价总结

Table 4 Evaluation summary of general detection model

文献	ML 算法	模型评价			
		Accuracy	Precision	Recall	F1_score
[15]	XGBoost	0.963	--	--	0.96
[50]	注意力机制	--	0.963	0.978	0.971
[14]	j48	0.9859	0.984	0.986	0.984
	RF	0.9816	0.982	0.982	0.974
	KNN	0.9877	0.986	0.988	0.987
	LR	0.747	0.718	0.816	0.764
	SVM	0.775	0.71	0.835	0.773
[47]	RF	0.995	0.99	1	0.995
	Sck	0.998	0.997	1	0.998
	Ada	0.998	0.997	1	0.998
[48,49]	RF	--	0.96	0.96	0.96
[51]	RF	--	0.96	0.96	0.96
[52]	LR	0.942	0.944	0.94	0.942
[53]	DT	--	0.95	0.7916	
	Gaussian NB	--	--	0.9858	

捉智能合约字节码中的隐藏信息, 再使用 N-gram 方法提取字节码特征。Baek 等[51]利用高斯混合模型的检测算法来设计钱包的九个特征, 包括交易特征以及时间戳的均值、方差、标准差等结构性特征。文献[47-49,52-53]注意将时间信息融入整体特征中。Poursafaei 等[47]提取了包括一般、邻域、局部和时间戳相关特征, 且应用插补、标准化和主成分分析等方法对数据进行预处理和清洗, 以获得干净和标准化的特征; Agarwal 等[48-49]使用时间粒度捕捉交易数据的行为变化, 执行时间序列分析来识别最能代表相关时间序列显著属性的特征; Sachan 等[53]通过分析 DNS 流量获得时间特征; 而 Poursafaei 等[52]采用带有负采样的 Skip-Gram 模型学习节点表示, 然后在下游任务中使用生成的节点表示来对非法和真实节点进行分类(逻辑回归)。

3.4 异常检测

利用提取出的特征因素, 研究者使用不同的 ML 算法进行异常检测工作。

如图 5, 第一种是先将地址聚类成具有相似行为模式的聚类簇, 再利用已知的标记信息进行身份判断和异常检测[17,46]。Morishima[17]使用 K-means 算法聚类, 同时通过设置两类阈值用于异常检测。第一个是顶点的特征量与包括该顶点的簇的重心之间的距离; 第二个是包含顶点的簇大小——如果距离超过阈值或远小于阈值, 则判断顶点异常。Sun 等[46]使用 Birch 算法(基于层次聚类的算法)对节点进行无监督

聚类。对不同的聚类簇, 借用已知的身份节点判断聚类簇的类型, 进一步基于节点的向量空间距离来检测恶意用户。

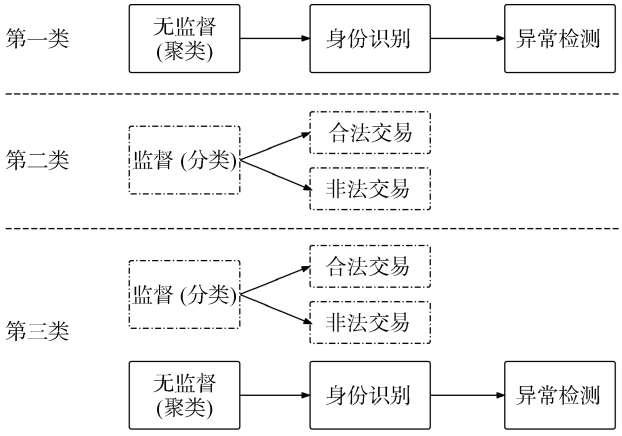

图 5 异常检测阶段的三类检测方法

Figure 5 Three types of detection methods in anomaly detection stage

第二种方法则是直接利用带标签的数据训练分类模型(监督算法)来判断账户的合法性[14-15,47,50]。Ibrahim 等[14]比较决策树(Decision Tree, DT)中采取自上而下的递归策略 j48 算法、随机森林(Random Forest, RF)算法和 K-近邻(K-Nearest Neighbor, KNN)算法三种监督算法, 发现 j48 算法在准确性方面表现最佳。Farrugia 等[15]训练 XGBoost 分类模型来识别数据背后的身份信息。Poursafaei 等[47]使用逻辑回归(Logistic Regression, LR)、支持向量机(Support Vector Machine, SVM)、随机森林三种分类模型和 Stacking Classifier(Sck)、AdaBoost(Ada)两种集成方法将实体分类为非法实体和合法实体。最终集成方法表现出最高性能, 平均 F1 得分为 0.996。Hu 等[50]使用神经网络框架 SCSGuard 进行检测任务, 使用合约字节码来检测智能合约中的诈骗。最终对于庞氏骗局的检测精度为 0.922, 蜜罐检测精度为 0.947。

第三种方法将监督算法和无监督算法结合[48-49,51,53]。Baek 等[51]利用 K-means 算法将钱包聚类为 7 个独立的集群, 再观察集群中已知身份信息的节点, 结合随机森林 (RF) 对钱包进行分类, 带有“1”标签的钱包表示异常钱包。文献[48,53]使用监督算法构建高精度检测模型, 另用无监督算法寻出潜在嫌疑人。Agarwal 等[48]发现 ExtraTrees 分类算法在有监督的设置下表现最好, 并且对于不同的数据集配置在[87.2, 88.7] 范围内实现了平衡的准确度; 此外, 在无监督设置下, K-means 算法能够将最多 73.9%的已知恶意账户聚集在一起, 并识别出 554 个与恶意账户具有相似行为的嫌疑人。进一步地, Agarwal 等还使用生成对抗网络 (GAN)测试了机器学习算法在对抗性输入上的鲁棒性[49]。Sachan 等[53]一方面使用 AutoML 工具 TPOT[54]验证和测试在分类恶意 DN 方面表现最好的监督算法; 另一方面, 使用 K-means 算法得到集群, 再确定包含最多恶意 DN 的集群, 通过计算恶意 DN 和良性 DN 之间的余弦相似度给 DN 贴上标签来进行异常检测。

3.5 模型评价

3.5.1 模型性能比较

(1)准确率: 准确率可以直观反映模型的检测效果, 即能否将交易根据合法性正确分类。根据表 4, 在通用非法交易的检测模型中, Sck、Ada 两种集成方法达到最高的准确率(0.998); j48、RF、KNN 三种监督算法在欺诈交易的检测上准确率超 0.98; LR、SVM 等传统的机器学习算法表现欠佳(不超过 0.8)。

(2)精度: 精度表示预测为非法交易的数据集中真实非法交易的占比, 这凸显出非法交易不能被错误预测。根据表 4, 除 LR、SVM 的检测精度不超过 0.8 外, 其余算法的检测精度均超过 0.95。其中 Sck、Ada 两种集成方法同样达到最高的精度(0.997)。

(3)召回率: 召回率表示真实非法交易数据集中正确预测量的占比, 体现非法交易检测模型的成功率。根据表 4, RF、Sck、Ada 达到 1, 说明数据集中所有的非法交易都被成功检测; j48、KNN 依然有高达 0.98 的召回率; LR、SVM 的召回率略高于准确度和精度, 达到 0.8 以上, 但依然低于 0.9。

(4)F1 分数: F1 分数是对准确率和召回率的均值, 同样可看出表现最佳的是 Sck、Ada 两种集成方法(0.998), 其次是 j48、RF、KNN 三种监督算法(0.98 以上), LR、SVM 机器学习算法依然在最后一个档次(不超过 0.8)。

综合上述模型评价, 可得出以下结论:

算法比较: 在诸多机器学习算法中, LR、SVM 等传统的机器学习算法表现欠佳, 这是由于以太坊的数据呈现高维异构网络形态, 根据节点距离划定分界线实现分类(合法交易与非法交易)的方法过于简单, 放弃了挖掘节点之间的深刻关系。j48、RF、KNN 三种监督算法的性能表现处于第二梯队, 因为 j48 与 RF 均属于决策树算法, 利用信息熵进行判断, 即加入了随机性, 而 KNN 是将特征空间中表现相似的分为一类, 考虑了特征的多维性, 上述三种算法能够有效提高检测效果。表现最佳的是 Sck、Ada 两种集成方法, 非法交易检测覆盖率达 100%, 这是因为一种算法有其局限性, 多种算法的集成不容易过拟合, 支持多种分类模型构造学习器, 实现更高水平的预

测。但是集成算法需要牺牲算力和复杂度, 这面对海量以太坊交易数据是不小的挑战。

针对 RF: 观察表 4, 发现文献[14,47-49,51]中均使用 RF 算法对数据集进行分类, 模型得分均在 0.96 以上。一方面, 高得分体现了 RF 算法在非法交易检测的优越性; 另一方面, Poursafaei 等[47]基于 RF 创建的检测模型得分最高(准确率 0.995, 精度 0.99, 召回率 1), 回溯其创建流程, 发现算法提取的特征更为全面(交易、邻域、本地、时间四个方面的特征)。这更加凸显特征提取在检测环节的重要性, 同样验证了 Poursafaei 等[47]特征提取角度的正确性, 即不仅关注时间的动态演变、交易的金额信息, 还关注网络结构, 重视节点的子图特征。

针对 LR: 表 4 中出现两次 LR 算法[47,52], 均由 Poursafaei 等使用, 但评分却呈现较大差距。在文献[47]中, LR 的准确率为 0.747, 与文献[52]中的(0.942)相差约 0.2, 其他分值同样呈现该差距。主要原因在于两篇文献的侧重点不一致——文献[47]重在对比多种分类器的使用效果; 文献[52]则关注基于图的非法检测方法。结果证明基于图的特征提取方法可以显著提高检测性能。

3.5.2 模型难点分析

研究发现通用的非法交易检测模型在不同阶段各有其重难点。

在数据收集阶段的难点体现在数据量庞大、对计算机的运算存储能力有高要求; 同时收集非法数据集也是一项耗费人力物力的工作, 但使用同一个非法数据集进行研究, 容易导致模型的过拟合问题。主要需求为庞大数据集的处理(Morishima[17]使用子图来减轻运算量), 以及社会面及时统计、公布已发现的非法交易, 搭建全球范围内的信用联动平台。

在异常检测阶段, 相较于无监督学习模型, 监督学习模型表现出更好的性能, 这是由于标记数据集的存在有利于对模型参数的训练和修正。而使用有监督算法和无监督算法相结合能够达到 98%以上的准确率。在有监督算法中, KNN 算法[14]和 ExtraTrees 算法[48]的性能最佳, 而集成方法能够综合多个算法的优势, 同样得到了很好的检测效果[47]。在无监督算法中, 大部分的研究均采用了传统的 K-means 聚类算法, 结合阈值、空间距离等其他方法帮助进行非法检测, 实现对未知非法交易的发现。两者结合可以最大限度地发现以太坊中潜藏的非法交易。

4 以太坊非法交易的特殊检测技术

伴随智能合约的特殊性, 犯罪活动迁移至以太坊上衍生出了“新形式”, 以下将主要讨论主要非法交易的新变化、新特点, 包括网络诈骗、以庞氏骗局为内核的传销、蜜罐合约等。

4.1 网络诈骗

自网络金融业务兴起, 网络诈骗成为交易安全的主要威胁。根据 Chainalysis 的报告[5], 2021 年涉及虚拟货币的犯罪达到了达到了创纪录的 140 亿美元。其中, 诈骗案件造成的损失高达 78 亿美元, 同比增长了 82%。表 5 总结了虚拟货币诈骗的几种常见形式。

表 5 虚拟货币诈骗的“新形式”

Table 5 “New forms” of virtual currency fraud

诈骗类型	诈骗手法
1.伪造虚拟货币服务	1) 假冒交易欺诈: 通过为追求加密货币的用户提供极具竞争力的市场价格来欺骗用户。 2) 假钱包: 钱包服务允许用户管理、发送和接收虚拟货币。在这种情况下, 用户可能会遭遇各种钱包诈骗。 3) 假混币器: 假混币器收到钱, 然后在不把钱寄给客户的情况下偷走。 4) 伪造挖掘池: 要求用户通过投资购买采矿硬件来参与挖矿奖励。 5) 假捐赠: 伪造捐赠活动欺骗赠金。
2.预付费诈骗	用假电子邮件地址或社交媒体帐户与受害者联系, 以承诺高回报来索取预付款。
3.勒索	发送邮件声称受害者设备已入侵, 并安装记录器, 以此威胁受害者索要货币。
4.虚假 ICO	通过伪造产品项目发行假币诱骗用户购买。

上述伪造相关服务进行诈骗的手段经常出现在各类虚拟货币中, 本质上通过发送虚假消息诱骗用户上当受骗, 属于网络钓鱼诈骗的变形, 以下以网络钓鱼诈骗为例分析新形式、新特点与新方法。

4.1.1 以太坊网络钓鱼诈骗

定义 1.网络钓鱼诈骗[55]. 网络钓鱼是一种计算机攻击, 它通过电子通信渠道将信息传达给受害者, 以说服他们为攻击者的利益执行某些操作。

传统场景下, 攻击者会依靠虚假网站和电子邮件获取用户的敏感信息或骗取财务。如图 6 所示, 主要分为 5 个环节进行网络钓鱼诈骗[56]: (1)侦察:寻找潜在客户群; (2)武器化: 设计钓鱼网站和垃圾邮件; (3)分发: 向受害者发送电子邮件; (4)入陷: 诱使受害者进入钓鱼陷阱; (5)收集: 从网络钓鱼数据库中收集敏感数据。

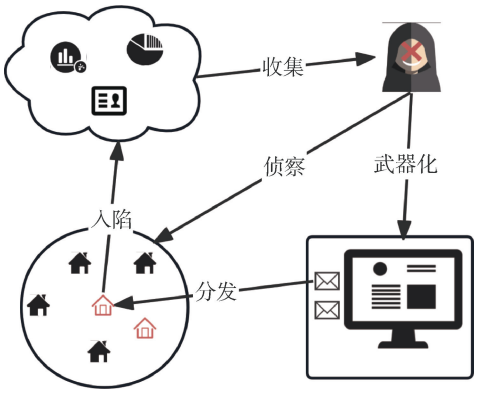

图 6　网络钓鱼诈骗流程

Figure 6　Phishing fraud process

与传统场景相比, 以太坊上的网络钓鱼诈骗在几个方面表现得非常不同:

(1) 攻击者的存在形式不同。传统场景设计钓鱼网站, 而以太坊上的攻击者主要以以太坊地址的形式存在。

(2) 攻击方式不同。传统场景是通过发送电子邮件诱骗受害者进入陷阱, 检测时有迹可循, 而以太坊上的信息是匿名的, 传播形式多样(详情参见表 3), 具有较高的隐匿性。

(3) 受害者的损失形式不同。传统场景下受害者有可能暴露个人敏感信息或失去财务, 而以太坊上的受害者失去的就是以太币。

(4) 数据的保存形式不同。传统的网络钓鱼交易记录较为分散; 而以太坊的所有交易记录都是公开可访问的, 这表示可以基于以太流寻找攻击者的蛛丝马迹, 实现对网络钓鱼的有效检测。

以太坊网络钓鱼攻击者重在“攻心”, 让受害者主动支付以太币, 对智能合约的要求不高, 故学者无法利用智能合约代码数据寻找蛛丝马迹, 只能尝试从链上交易数据中寻找相关线索。链上交易数据以交易网络的形式存在, 通常具有边缘属性: 时间戳与交易金额, 这些属性被视为检测关键。检测方法也随之发生变化:

(1) 检测对象不同。传统场景下主要检测钓鱼网站, 新形势下的目标对准网络钓鱼网站的以太坊地址。

(2) 检测信息不同。传统场景下关注网站的内容和 URL 信息, 例如 URL、超链接、暗示网络钓鱼可能性的敏感词, 以及基于电子邮件和 Web 内容的文本检测等内容信息; 以太坊上的检测则对准地址之间的公开交易记录来尝试识别出攻击者。

(3) 检测方法不同。传统场景下检测钓鱼网站主要包括基于规则、黑名单、网站内容在内的启发式方法以及基于机器学习的方法, 且集成方法能达到最佳检测效果。相较之下, 以太坊网络钓鱼缺少相关内容信息; 进一步地, 观察网络钓鱼的以太流, 可发现交易呈聚合性(多个账户地址向同一账户地址支付金额), 但该特点不存在唯一性——交易所同样满足该特点, 故无法使用启发式方法进行检测。以太坊网络钓鱼检测专注于使用基于机器学习的方法, 利用带标记数据训练检测模型, 提高对账户地址(节点)的分类效果。检测重点在于提取网络钓鱼的特征训练高准确度的检测模型, 难点在于数据集的极端不平衡以及网络异构化。

数据集的极端不平衡是以太坊网络钓鱼检测的主要问题之一。据 etherscan.io 报告, 以太坊的独立地址数已累计超 1.65 亿个, 而发布的钓鱼地址总数不超过 1 万个, 检测钓鱼地址犹如大海捞针。

以太坊交易网络的网络异构性是指大量交易由部分公开地址(如交易所、著名的 ICO)进行, 而普通地址和钓鱼地址的交易量较少, 导致钓鱼地址的网络拓扑特征不明显。故针对以太坊上的网络钓鱼检测问题, 使用手动提取特征、PCA 特征降维等常见的特征提取方式均无法得到高召回率的检测模型[57-58]。而图嵌入为解决异构网络问题提供了一种行之有效的方法。具体来说, 图嵌入将图转换为保留图信息的低维空间。通过将图表示为一个(或一组)低维向量, 可以有效地降低交易网络的数据维度, 将大规模稀疏的高维节点向量转化为密集的低维节点向量, 实现对多维数据的精准研究。近年来, 图嵌入技术在各种网络的分析中得到了广泛的应用[59-60], 例如社交网络[61]、网络安全[62]、生物和化学信息学[63]。图嵌入技术主要有四类[64-65]: 矩阵分解[66]、深度学习[67]、图内核[68]和其他方式。

诸多学者将基于图的研究成果应用到以太坊用户交易网络中, 不仅关注图的多级拓扑结构, 使用图嵌入技术进行特征提取, 而且加入时间因素, 构建动态事务图, 取得了高精度的检测效果[69-78]。

4.1.2　基于随机游走的图嵌入检测模型

文献[69-74]均使用深度学习中基于随机游走的图嵌入方法提取特征, 主要利用游走来感知节点的中心性和相似性(如表 6 所示)。

文献[69-71]尝试从交易静态图中进行特征提取。Yuan 等[69]引入有偏随机游走, 使用 node2vec 在广度优先采样和深度优先采样策略之间进行平滑搜索。文献[70-71]均使用 Graph2Vec 模型进行特征提取。主要步骤包括: (1)提取有根子图, 以便以一组子图的形式表示所有图; (2)通过 skip-gram 模型学习图

表 6 基于随机游走的以太坊网络钓鱼诈骗图嵌入检测模型

Table 6 Ethereum phishing fraud graph embedding detection model based on random walk

时间	作者	网络构建	图嵌入	分类模型	precision	recall	F1-score
2020	Yuan 等[69]	交易网络	node2vec (随机游走)	SVM	0.871	0.822	0.846
2020	Yuan 等[70]	交易网络	Graph2Vec (随机游走)	SVM	0.69	0.77	0.73
2021	Wang 等[71]	TNs, TSGNs, Directed-TSGNs	Handcrafted (人工)	RF	--	--	0.775
			Graph2Vec (随机游走)	RF	--	--	0.6815
			Diffpool (深度学习)	RF	--	--	0.9435
2020	Wu 等[72]	交易网络	trans2vec (随机游走)	SVM	0.927	0.893	0.908
2020	Lin 等[73]	时间加权多向图 TWMDG	T-EDGE (随机游走)	节点分类	--	--	0.84
2021	Xie 等[74]	TSSN	TBW (随机游走)	LR	--	--	0.8898

的嵌入。区别在于对交易信息的表达方式不同: Yuan 等[70]研究二阶交易子图, 将以太币流信息整合到 Graph2Vec 模型中; Wang 等[71]通过增加网络权重映射机制保留交易信息, 并且对比不同信息表征下的子图效果, 发现引入交易流信息构建的有向交易子图网络(Directed-TSGNs)性能最优秀。但静态子图方法往往会忽略不断变化的交易行为中的时间特征。

故文献[72-74]在交易图中加入时间因素。Wu 等[72]设计具有交易金额和时间戳的偏差的 trans2vec 算法, 将交易信息嵌入到节点表示向量中; 进一步的, Lin 等[73]使用时间加权多向图嵌入算法(T-EDGE)进行特征表示, T-EDGE 通过交易量和间隔扩展了随机游走中被访问的边缘概率。Xie 等[74]搭建时间序列快照网络(TSSN)将以太坊交易记录建模为时空网络, 提出利用时间偏差游走(TBW)来学习账户表示, 集成时间和结构信息。但随机游走方法因为有限的采样序列长度, 通常会丢失结构信息。

4.1.3 基于深度神经网络的图嵌入检测模型

另一类研究方法通过构建深度神经网络来学习图中的非线性信息[75-78]。深度神经网络(Deep Neural Networks, DNN)指具有多层参数和转换的神经网络, 通过深度学习技术有助于识别网络中的异常模式(如表 7 所示)。

表 7 基于深度神经网络的以太坊网络钓鱼诈骗图嵌入检测模型

Table 7 Ethereum phishing fraud graph embedding detection model based on deep neural network

时间	作者	网络构建	图嵌入	分类模型	precision	recall	F1-score
2020	Chen 等[75]	GCN	Autoencoder (深度学习)	LightGBM	0.7294	0.1735	0.2636
2021	Li 等[76]	交易网络	SIEGE (深度学习)	LR	0.456	0.68	0.546
2021	Zhang 等[77]	用户交易模式图	MCGC	多通道图分类	--	--	--
2022	Kanezashi 等[78]	交易子图	RGCN	节点分类	0.8958	0.7454	0.8124

Chen 等[75]统计每个节点的边缘信息作为节点的特征, 创建图卷积网络(Graph Convolutional Networks, GCN), 使用自动编码器技术提取结构特征, 最后使用 LightGBM 的检测方法来精确区分钓鱼账户。Li 等[76]采用自我监督学习(self-supervised learning, SSL)直接从数据中挖掘信息, 设计具有增量训练的自监督增量深度图学习模型(SIEGE)进行非法检测, 以解决标签稀缺性和数据可扩展性问题。Zhang 等[77]构建多通道图分类模型(MCGC)进行检测, 这是一种具有多通道图神经网络(Graph Neural Network, GNN)架构的图分类模型, 以可训练的方式聚合多个通道中的池化图信息, 从而实现更好的图分类性能。进一步地, Kanezashi 等[78]分别考虑同构 GNN 模型(节点和边均为单一类型)和异构 GNN 模型(节点和边均支持不同类型)在以太坊钓鱼检测上的性能表现, 实验结果表明后者更能精准预测。其中,

多关系异构图(RGCN)模型在整体指标上取得了最好的表现。

4.1.4 其他检测模型

文献[79-80]从交易网络的多级拓扑角度丰富图的结构信息(如表 8 所示)。Chen 等[79]提出了一种基于图的级联特征提取方法, 不仅提取该节点的数据特征, 还提取 n 阶邻居(连接到至少有 n-1 个节点的节点)的统计特征。然后结合基于交易记录和基于 lightGBM 的双采样集成算法来构建识别模型。Chen[80]同时保留时间信息和结构信息。主要方法是为一个目标地址构建一系列事务演化图(TEG), 其中每个子图表征一个时间段内的事务拓扑, 再利用动态图分类器(TEGDetector)捕捉网络的拓扑结构和动态演化特征。

表 8 其他基于图应用的网络钓鱼诈骗检测模型

Table 8 Other phishing fraud detection models based on graph application

时间	作者	数据获取	特征工程	分类模型	precision	recall	F1-score
2020	Chen 等[82]	Etherscan(双采样)	基于图的级联特征提取	双采样集成模型	0.8258	0.839	0.8324
2021	Chen 等[83]	Xblock	TEGDetector	Softmax 分类器	0.963	0.9625	0.9628

4.1.5 以太坊网络钓鱼诈骗检测模型比较

目前主流的网络钓鱼检测模型是基于图进行创建的, 但是性能却有较大差别。

(1)基于随机游走: 根据表 6, 可看出精度和召回率最高的模型是 trans2vec+SVM(精度 0.927、召回率 0.893), node2vec+SVM 次之, Graph2Vec+SVM 得分最低(低于 0.8); 三者之间的区别主要在于 trans2vec+SVM 考虑增加时间戳元素, 特征提取更全面, 从而获得更好性能。若关注 F1 分数, 最佳模型变为 Diffpool+RF(0.9435), trans2vec+SVM 退居第二(0.908), 其余模型均在 0.9 以下。其中 Diffpool+RF 使用深度学习的方式提取特征, 且 TSGN 可以保留拓扑模式的交易流信息, 充分体现出深度学习的优越性。另一方面, 结合 TBW+LR 的 F1 分数(0.8898), 发现若使用合适的特征提取方法, 利用 SVM 和 LR 等传统的分类器同样可以达到较好的检测效果。

(2)基于深度神经网络: 根据表 7, 发现没有模型得分超过 0.9。观察整体表现, RGCN 模型性能最佳(精度与 F1 分数均超过 0.8), 其构建的网络中不仅包含异构信息, 还挖掘了特征之间的关系; MCGC 仅包含准确率数据(0.81), 体现出一定的分类能力, 但无法与其他模型比较; Autoencoder 与 SIEGE 性能不完备, 前者呈现出较高精度(0.7294)低召回率(0.1735)与低 F1 分数(0.2636)的特点, 后者却是低精度(0.456)中召回率(0.68)中 F1 分数(0.546)。精度与召回率的区别在于分母集的不同, 精度选定的是预测集、召回率选定的则是真实集。这是由于 Autoencoder 遵循疑罪从无规则, 即检测出的非法交易需保证其真实性, 这导致数据集中仍有大量真实非法交易未被发现; SIEGE 则遵循宁可抓错不可放过原则, 表现为检测出的非法交易的只有一半确为真实的, 但是大部分的真实非法交易都被检测出来了。如果综合双方技术, 或许可以取得更高的精度和召回率。

(3)其他基于图的应用: 根据表 8, 文献[79-80]的模型性能整体表现良好, 其中 TEGDetector 在所有网络钓鱼检测模型中表现最佳(精度、号召率和 F1 分数均超过了 0.96), 归功于 TEGDetector 可以同时捕捉网络拓扑结构和动态演化特征, 分类依据十分充实; Chen 等[79]采用双采样集成算法, 模型评分均在 0.8 以上, 表现上佳, 若增加考虑时间信息, 或许能进一步提升模型性能。

(4)综合评价: 综合上述模型, 可发现融合结构+时间信息的特征提取方式能够显著提高检测性能(例如 trans2vec+SVM、TEGDetector); 基于随机游走与深度神经网络的检测模型性能差异较大, 但是前者存在评分超过 0.9 的检测模型(trans2vec+SVM), 后者的评分均在 0.9 以下, 需要学者们进一步深入挖掘以太坊网络和数据结构等特征, 并融合深度神经网络算法; TEGDetector 采用动态图分类思想实现最佳检测效果, 对特征提取作出新的探索。

研究发现, 图嵌入算法用于以太坊网络钓鱼检测能够得到更高的准确率和召回率, 比单一的图拓扑分析的结果好很多。而时间序列、二阶交易子图、图的级联特征等思维角度的加入也有利于特征工程的进一步优化, 从而丰富特征信息。而如何保证检测模型应用到以太坊上依然能保持高检测率、即保持模型的鲁棒性, 还有很大的研究空间, 有待研究者们去探索发现。

4.2 以庞氏骗局为内核的传销

4.2.1 以太坊传销

以以太坊为代表的虚拟货币型传销就是指传销分子将虚拟货币作为工具或者以虚拟货币为噱头, 以投资虚拟货币高额返利为诱饵, 借助互联网平台拉人头形成层级关系, 以发展下线的数量和投资虚

拟货币的价值为返利依据的一种网络传销。

犯罪流程为: (1)推广项目, 发展人员; (2)利用静态收益, 诱导投资; (3)利用动态收益, 拉人返利; (4)延缓提币, 关门跑路。

以太坊上常见的传销模式有(如表 9 所示):

(1) 质押挖矿模式: 犯罪分子打着“通过挖矿奖励虚拟货币”的旗号, 吸引参与者在交易所购买指定的主流币后, 放在犯罪嫌疑人这里质押。

(2) 合约交易所模式: 犯罪分子建立山寨合约交易所后, 以金字塔式返佣为手段拉人头, 庞氏骗局就是一种典型的投资传销模式。

(3) 理财钱包模式: 犯罪分子打着理财幌子吸引受害者参加, 以发展下线数量和钱包充值金额作为返利依据, 实现层级联系。

(4) DeFi 模式: 有别于前三种传销模式, DeFi 传销模式没有涉案 APP, 也不是通过邀请码产生层级关联。具体而言, 嫌疑人首先需要一个地址将涉案的智能合约部署到区块链上, 然后在项目社群中推出一个 DeFi 项目, 许诺高额收益。参与者会通过社群获取网址, 在主流钱包中搜索该网址, 进入 DAPP 页面, 获取到主流钱包地址, 该地址就是该参与者的唯一身份标识。通过输入上下线地址, 形成层级关系, 参与者将虚拟货币从交易所或钱包地址中转到该涉案地址上, 嫌疑人控制后卷币跑路。

表 9 以太坊传销的几种典型模式

Table 9 Several typical models of Ethereum MLM

模式	项目噱头	层级关联
质押挖矿	通过挖矿奖励虚拟货币	发展下线数量和质押币金额决定返利金额
合约交易所	建立合约交易所	金字塔式返佣金
理财钱包	创建虚拟货币理财钱包, 提供数字货币增值服务	参加者需要上线推荐取得会员账号
DeFi	推出 DeFi 项目, 许诺高额收益	输入上下线地址, 形成层级关系

总结发现传销的内核在于利用新投资者的资金向现有投资者支付回报, 即庞氏骗局。

4.2.2 以太坊庞氏骗局

美国证券交易委员会(SEC)对庞氏骗局的权威定义[81]如下:

定义 2.庞氏骗局. 庞氏骗局是一种投资欺诈, 涉及从新投资者提供的资金中向现有投资者支付所谓的回报。庞氏骗局的组织者经常通过承诺将资金投资于声称可以产生高回报且风险很小或没有风险的机会来招揽新的投资者。由于收益很少或没有合法收益, 庞氏骗局需要来自新投资者的持续资金流才能继续。庞氏骗局不可避免地会崩溃, 最常见的是当招募新投资者变得困难或大量投资者要求退还他们的资金时。

通常庞氏骗局用户拓扑结构呈金字塔形, 在顶层的发起人用第 l+1 层用户的资金补偿第 l 层的用户(如图 7 所示)。该计划最终会崩溃, 因为在某个时候, 随着投资者数量在金字塔的层数中呈指数级增长, 将不再可能找到新的投资者, 而那些处于底层的人只会失去他们的投资。在此基础上, 一些犯罪分子对庞氏骗局进行包装, 发展出了传销模式。

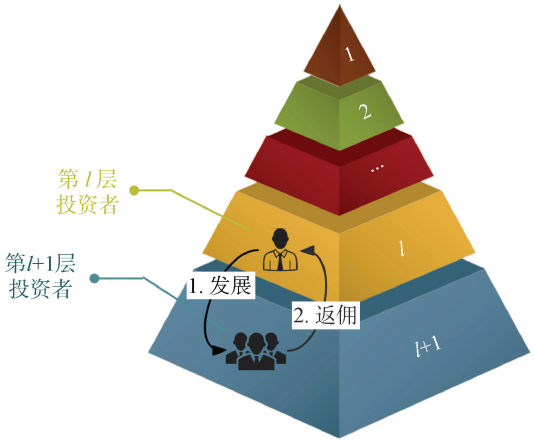

图 7 金字塔形庞氏骗局流程

Figure 7 Pyramid Ponzi scheme process

可以看出, 传销就是经过包装的庞氏骗局, 在庞氏骗局的核心基础上多了“项目”和复杂的“层级联系”, 但交易内核未发生改变。故文章以庞氏骗局为代表, 进行交易检测。

以太坊上庞氏骗局主要存在两种行为[82]: 1)投资行为, 即用户调用交易用以太币进行投资, 并且在智能合约中保存用户信息; 2)奖励行动, 庞氏骗局通过金字塔策略为参与者支付奖金。只有当有足够多的受害者落入陷阱时, 庞氏骗局才有足够的钱来奖励参与者。因此, 庞氏陷阱必须在生命周期内保存参与者的信息, 以执行奖励行动。根据其重新分发策略和用于存储用户信息的数据结构, 将庞氏骗局智能合约进一步分类为四种类型, 包括切换方案、链方案、树方案和撤回方案(如表 10 所示)。

(1) 切换方案。在这个方案中, 最底层玩家将获得新参与者携带的所有资金。金字塔中始终只有一个投资者: 一旦新玩家投资, 接收者的地址将存储在一个状态变量中, 合同将为其支付费用; 在奖励操作之后, 它将接收方变量设置为调用者, 表示合同将特权移交给新的投资者。而投资活动期间, 合同所有者可随时提取 10%的费用。

表 10 以太坊上庞氏骗局的四种方案

Table 10 Four schemes of Ponzi scheme on Ethereum

方案	参与者结构	奖励机制
切换方案	孤立点	每位参与者投资的资金将交给上一位参与者, 且从下一位参与者处获得双倍资金。
链方案	链式结构	每位参与者对应一个序号, 新玩家携带的资金按照序号分配给老玩家, 且金额数按序号递减。
树方案	树状结构	若新玩家 B 由老玩家 A 发展, 则老玩家 A 可获得最高佣金。
撤回方案	股东结构	新玩家携带的资金会按照投资比例分配给老玩家。

(2) 链方案。该方案使用线性数据结构来维护投资者的信息。每位参与者对应唯一一个序列号(通常按参与顺序递增)。然后, 在奖励阶段, 将根据序列号重新分配投资。一般来说, 在这个计划中, 加入的时间越早, 获利越高。

(3) 树方案。与链方案类似, 区别在于玩家结构呈树状。类似于绩效奖励, 一旦“老”玩家 A 邀请新玩家 B, 奖励会从老玩家 A 开始发放, 遍历直到根节点。每次分发后, 付款金额将减少一半。因此, 老玩家 A 或 A 的子节点邀请的玩家越多, 获利越多。

(4) 撤回方案。玩家通过将以太币直接传输到智能合约参与该游戏。类似于股东制, 新玩家的以太币将根据投资比例分配给老玩家。之后, 新玩家将被追加为投资者, 并等待下一个参与者的投资。此外, 通过退出功能, 投资者可以自由提取他们的利润。该方案下的合同是以太池, 通过投资扩大并分配给参与者。

传统的庞氏骗局可以人为终止, 其组织者可以随时随钱消失。但在以太坊中, 庞氏骗局运转规则往往以智能合约的形式存在, 一旦开始执行则无法人为终止, 除非满足智能合约代码中的预设条件。事实上, 与传统场景相比, 在以太坊上运转庞氏骗局具有以下新特点:

(1) 匿名性。庞氏骗局的发起人可以全程利用以太坊账户隐藏身份。

(2) 不可更改性。由于智能合约一旦运行无法手动修改和终止, 任何政府都无法终止该计划的执行, 这导致金额无法被追回。另一方面, 受害者还会因为智能合约代码的公开和自动运行产生虚假信任——计划将会永远执行下去, 即他们会一直接收佣金。

(3) 隐蔽性。庞氏骗局的交易量在以太坊上总成交量中所占比例非常小, 即在海量交易中, 庞氏骗局不易被发现。

(4) 公开性。以太坊上的智能合约和交易都会在链上保存, 所以可以通过总结智能合约和交易特征发现庞氏骗局案例。

文献[82-84]总结庞氏骗局的“智能合约”特征:

(1) 要求智能合约将资金分配给投资者, 即通过向其发送一些资金来加入合约的用户。

(2) 要求合约收取的资金仅来自投资者。这排除了分配给投资者的资金来自外部来源的情况, 比如支付债券利息的银行。

(3) 要求每个投资者都能盈利, 前提是新投资者继续向合同提供资金。

(4) 要求失去投资的风险随着加入时间而增加。

(5) 合约中存储用户信息, 例如投资者的投资与接收者的地址。

(6) 合约在生命周期内保存参与者的信息, 保证有足够多的新投资者加入, 才能有足够多的资金奖励上一层的玩家。

(7) 合同的余额金额可能很低, 因为庞氏骗局总是试图保持快速高回报的形象。

而以太坊上庞氏骗局的交易行为特征[84-86]有:

(1) 支付交易通常发生在投资交易之后, 这表明合同通常支付给已知账户。

(2) 许多投资交易没有后续支付交易。

(3) 一些参与者的支付交易多于投资交易。

(4) 不平等的资金分配会体现出较高的基尼系数。

(5) 少数合同参与者的回报率很高, 基本上所有的回报都集中在一两个参与者(创建者)身上。

针对以太坊上庞氏骗局的行为模式, Bartoletti 等[83]首次将庞氏骗局分为四种方案, 即基于数组的金字塔方案、基于树的金字塔方案、切换方案和瀑布方案。在此基础上, 后期研究集中在利用机器学习算法进行庞氏骗局检测, 重心在于数据处理、特征提取和 ML 算法阶段。

但学者们的研究角度略有不同。部分学者重视对庞氏骗局的全盘把握, 提取包括交易信息和智能合约代码信息在内的全部特征, 建立全特征模型[82,84-90]来检测庞氏骗局; 同时, 亦有学者关注到庞氏骗局智能合约的不可逆性, 仅使用智能合约的代码信息建立 0-day 模型[84-85,91-93], 将检测提前到智能合约的部署阶段并进行事前预警, 保障用户权益。接下来分别对全特征模型和 0-day 模型进行介绍。

4.2.3 庞氏骗局的全特征检测模型

为了实现对以太坊上正在运行的庞氏骗局的检测, 部分学者综合利用以太坊账户的交易信息和智能合约代码信息(包括字节码、操作码以及应用程序

二进制接口(Application Binary Interface, ABI)调用等信息)来训练自己的检测模型, 该类模型即被称为全特征模型。(如表 11 所示。)

诸多学者关注特征提取的全面性。Chen 等[84]对比 Rubixi(一种典型庞氏骗局)和 LooneyLottery(普通彩票游戏合约), 首次从智能合约的交易历史和操作码中提取特征, 然后基于三类特征——“交易”“操作码”“交易+操作码”训练 XGBoost 分类模型来检测庞氏骗局。实验结果表明, “交易+操作码”模型性能最佳(精度为 0.94, 召回率为 0.81), 并在此基础上进一步扩展数据集的交易特征[85]。Jung 等[86]扩展行为特征选择, 将模型精度提高到 0.98。Zhang 等[88]提取字节码特征, 结合用户交易和操作码频率, 得到更全面的特征。

针对特征提取困难问题, Zhang 等[88]使用互斥特征绑定(Exclusive Feature Bundling, EFB)、直方图加速以及一种基于梯度的单侧采样(GOSS)算法, 将交易、字节码和操作码三种类型的特征结合起来。Liang 等[89]关注到特征提取需要耗费大量的人力资源, 提出一种基于动态节点嵌入技术的智能庞氏骗局检测系统。Yu 等[90]使用图嵌入方法, 将结构信息与节点特征相结合形成图卷积网络(GCN)[94]并提取特征。

表 11 庞氏骗局的全特征检测模型

Table 11 Full-feature detection model of Ponzi scheme

时间	作者	特征提取		ML 算法	precision	Recall	F1_score
2018	Chen 等[84]	交易特征		XGBoost	0.74	0.32	0.44
		操作码特征		XGBoost	0.90	0.80	0.84
		交易特征 +操作码特征		XGBoost	0.94	0.81	0.86
2019	Chen 等[85]	交易特征 + 操作码特征	传统分类	IF	0.02	0.05	0.04
				OCSVM	0.05	1.00	0.10
			集成方法	OCSYM+DT	0.33	0.21	0.25
				OCSVM+SVM	0.91	0.16	0.27
				SVM	0.91	0.16	0.27
			机器学习	DT	0.31	0.24	0.27
				XGBoost	0.90	0.67	0.76
				RF	0.95	0.69	0.79
2019	Jung 等[86]	操作码特征		J48	0.97	0.93	0.95
				RF	0.96	0.96	0.96
				SGD	0.98	0.94	0.96
		交易特征		J48	0.93	0.87	0.90
				RF	0.98	0.84	0.91
				SGD	1.00	0.08	0.15
		交易特征 +操作码特征		J48	0.98	0.97	0.97
				RF	0.93	0.92	0.93
				SGD	0.99	0.81	0.86
2021	Wang 等[87]	交易特征 +操作码特征		LSTM	0.97	0.96	0.96
2021	Zhang 等[88]	交易特征 +操作码特征 +字节码特征		LightGBM	0.967	0.967	0.967
2021	Liang 等[89]	交易特征 +操作码特征		DSPSD	0.98	0.85	0.91
2021	Yu 等[90]	交易特征 +操作码特征		RF(基于 GCN)	0.8564	0.9409	0.8963

在 ML 算法的选择中, 文献[85-90]有着不同偏好。Chen 等[85]对比三种类型的检测算法, 发现传统分类算法的检测效果很差, 而 RF 性能最佳。Jung 等[86]分别使用 J48 决策树、RF、随机梯度下降(Stochastic Gradient Descent, SGD)作为分类器, 均可实现良好检测。Wang 等[87]使用训练集来训练用于庞氏骗局智能合约检测的长短期记忆(Long Short-Term Memory, LSTM)模型, 再使用 Sigmoid 激活函数判断

智能合约是否为庞氏骗局合约。Zhang 等[88]提出一种基于改进的LightGBM的检测模型, 用Smote_Tomek混合采样代替LightGBM权重分配。Liang 等[89]将多模态信息整合为低维向量, 输入到由多层感知器组成的分类器中进行检测。Yu 等[90]注意交易网络的拓扑结构, 提出基于图卷积网络的检测模型来识别庞氏骗局智能合约, 召回率高达0.94。

4.2.4 庞氏骗局的0-day检测模型

虽然庞氏骗局的交易数据有助于训练检测模型, 但这意味着基于交易数据训练的检测模型仅在庞氏骗局发生后才能发挥作用, 此时用户损失已经产生。如果庞氏交易智能合约部署的一瞬间就能被检测出来, 就有可能实现零损失。由于智能合约代码特性在合约上传到区块链后立即可用, 使用代码特性构建的0-day模型就可以实现事前预警。已探索出的智能合约代码特征包括操作码特征[85]、字节码特征[82]以及合约调用时的ABI特征[93]。

受限于特征来源的单一性, 有学者从技术角度寻找合适的机器学习算法; 还有学者从数据角度尝试寻找更多潜在特征, 从而提高检测模型的精度。

如表12, Chen 等[84]仅基于智能合约的操作码特征训练的XGBoost模型检测效果尚佳; Jung 等[86]发现利用SGD机器算法可达到精度为0.98, 召回率为0.96的检测效果; Fan 等[91]使用有序提升的思想进行训练, 直接处理类别特征, 避免由目标泄露引起的预测偏移; Sun 等[92]利用行为森林构建了操作码运行树来模拟合约的动态执行, 然后采用自适应的全路径树编辑距离算法计算相似度实现检测。除了关注操作码特征, BIAN 等[93]还提取字节码序列和ABI调用信息作为特征, 检测模型(SE-CapsNet)被证明可以在以太坊上有效地检测庞氏骗局合约。针对特征提取的合理性问题, Chen 等[82]严格遵循庞氏骗局定义, 提出一种启发式引导的符号执行技术, 从而确保模型的可扩展性和可解释性。实验结果表明, SADPonzi可以达到100%的精度和召回率。

另一个关注焦点是类不平衡问题。庞氏骗局交易与整个以太坊交易相比犹如沧海一粟, 导致检测模型过拟合和泛化能力弱。部分文献通过数据增强方法尝试缓解该问题[87-88,91,93]。由于边界上的合约和附近的合约比远离边界的合约更容易被错误分类, Wang 等[87]和Fan 等[91]均使用Borderline-SMOTE 2过采样技术作为数据增强方法以实现更好的预测。Zhang 等[88]则用Smote_Tomek混合采样取代LightGBM的权重分配。BIAN 等[93]将图像增强技术应用到以太坊的数据集中, 比较诸多图像增强方法, 最终发现花式PCA方法更有助于实现高精度的分类结果。

4.2.5 以太坊庞氏骗局检测模型比较

以太坊庞氏骗局检测模型主要分为全特征模型与0-day模型两种, 以下进行类内和类间比较分析。

表12 庞氏骗局的0-day检测模型

Table 12 0-day detection model of Ponzi scheme

时间	作者	特征提取	ML算法	precision	Recall	F1_score
2021	Chen 等[82]	操作码特征	SADPonzi(语义感知)	1	1	1
2018	Chen 等[84]	交易特征	XGBoost	0.74	0.32	0.44
		操作码特征	XGBoost	0.90	0.80	0.84
		交易特征+操作码特征	XGBoost	0.94	0.81	0.86
2019	Jung 等[86]	操作码特征	J48	0.97	0.93	0.95
			RF	0.96	0.96	0.96
			SGD	0.98	0.94	0.96
		交易特征	J48	0.93	0.87	0.90
			RF	0.98	0.84	0.91
			SGD	1.00	0.08	0.15
		交易特征+操作码特征	J48	0.98	0.97	0.97
			RF	0.93	0.92	0.93
			SGD	0.99	0.81	0.86
2020	Fan 等[91]	操作码特征	PonziTect	0.98	0.97	0.98
2020	Sun 等[92]	操作码特征	BF	0.946	0.930	0.937
2020	BIAN 等[93]	字节码特征+操作码特征+ABI调用特征	SE-CapsNet	0.9779	0.9898	0.9838

针对全特征检测模型, 根据表 11 发现以下特点:

(1)机器学习算法应用于非法检测的有效性。观察整体情况, 发现 XGBoost、RF、J48、SGD、LSTM、LightGBM、DSPSD 等算法的检测评分大部分超过 0.9, 其中精度、召回率和 F1 分数最高的分别为 SGD(0.99)、J48(0.97)、J48(0.97), 可看出 Jung 等[86]创建的模型值得参考与研究; IF 与 OCSYM+DT 的整体表现很差(低于 0.4); OCSVM 与 OCSVM+SVM 功能单一, 前者牺牲精度实现完美召回率(精度 0.05, 召回率 1)、后者恰好相反(精度 0.91, 召回率 0.16), 即使用集成方法也无法全面提高 OCSYM 的模型性能。这是因为以太坊上交易数据量异常庞大, 而庞氏骗局数据犹如九牛一毛, 导致数据集极端不平衡, 使得通过确定边界来划分数据的 IF 和 SVM(包括 OCSYM)难以工作。对应的, ML 算法能利用测试集不断训练模型参数, 充分利用数据信息, 从而实现精准预测。

(2)数据特征的全面性有利于提高检测精度。Chen 等[84]与 Jung 等[86]对比“交易”“操作码”“交易+操作码”三类特征训练的检测模型性能, 发现“交易+操作码”的性能最佳。Zhang 等[88]首次结合交易、字节码和操作码特征, 利用 LightGBM 实现有效检测(精度、召回率和 F1 分数均为 0.967)。对比 Chen 等[84]与 Jung 等[86]的最佳模型, 前者的分类器为 XGBoost (精度 0.94、召回率 0.81、F1 分数 0.86), 后者的为 J48(精度 0.98、召回率 0.97、F1 分数 0.97), 可看出后者的性能更优。这是由于 XGBoost 重点考虑包括 SSTORE、SLOAD 和 CALLDATALOAD 在内的操作码, 但这些操作码广泛用于其他类型的智能合约, 所以不能代表庞氏骗局的关键特征; J48 采用贪婪和自上而下的决策树方法, 通过测试每个属性来计算信息增益实现分类, 尽可能全面的获取具有代表性的特征。

针对 0-day 检测模型, 根据表 12 发现以下特点:

(1)0-day 检测的有效性。虽然 0-day 检测模型数据来源相对单一(仅包含智能合约数据), 但是性能评估得分均超过 0.9(除 XGBoost), 这表示仅靠智能合约数据依然可以有效识别庞氏骗局。另一方面, 观察 Chen 等[84]与 Jung 等[86]的研究结果, 发现“交易+XGBoost”明显逊色于“操作码+XGBoost”(后者比前者每项评分均高出至少 0.15); “交易+J48\RF\SGD”的性能也不如“操作码+J48\RF\SGD”的性能全面(后者每项评分均超过 0.93, 前者评分颇有起伏)。从中可看出智能合约的操作码特征对于识别以太坊上庞氏骗局的贡献率更大, 即智能合约特征更具代表性。

(2)语义感知方法的优越性。观察表 12, 发现 SADPonzi 实现了 100%的精度、召回率和 F1 分数, 性能十分强大。分析 SADPonzi 的运行逻辑, Chen 等[82]首先为智能合约中的每个可行路径生成语义信息, 然后尝试匹配给出的行为模式, 最终判断该行为是否为庞氏骗局。因为智能合约是人为编写的, 通过分析语义感知行为是行之有效的方法。而 XGBoost、SGD、PonziTect、BF、SE-CapsNet 等算法更注重从数据中挖掘结构特征, 忽略了庞氏骗局创建思路的分析, 故评估得分略有不及。但是由于使用的数据集不同, 不能单纯判断 SADPonzi 的性能优于表 12 中其他模型, 需客观对待每个模型做出的贡献。

最后综合两种模型, 发现在庞氏骗局检测模型中, 基于机器学习构建的模型几乎能实现 0.95 以上的精度, 这证明了机器学习的有效性和准确性。但数据收集阶段和提取特征阶段采用的方法不尽相同, 导致准确率和召回率有着些许的差别。综合比较所有的全特征模型, Jung 等[86]和 Liang 等[89]构建的模型准确率均达到 0.98, 表明挖掘交易的行为特征、使用 j48(决策树)模型能够实现精确检测[86]; 而深度学习和动态节点嵌入技术能够代替传统的人工提取特征方法, 大大减轻了工作量。而在 0-day 模型方面, Fan 等[91]使用过采样技术平衡数据集, 训练模型的过程中利用了有序提升的思想, 实现了 0.98 的准确率; 而 Bian 等[93]的模型召回率达到 0.9898, 其挖掘出合约中包括源代码在内的更多数据信息, 可以避免错过潜在的庞氏骗局。

故在数据收集阶段, 关注到类不平衡问题, 可使用混合采样、过采样等数据增强方法来平衡数据集。在特征提取阶段, 注意考虑时间因素和结构因素, 着重挖掘更多的更有代表性的特征, 实现更高的召回率。在检测阶段, 传统的异常检测模型都不适用于庞氏骗局, 而基于深度学习的机器学习算法在庞氏骗局检测方面具有很大的潜能; 同时不能忽视庞氏骗局智能合约的创建逻辑, 借助符号分析功能能够实现全面检测。

4.3 蜜罐合约

在过去几年中, 攻击者主动寻找智能合约的漏洞进行攻击。近期, 一种更隐蔽的新方法似乎正在兴起, 攻击者不再搜索易受攻击的合约。相反, 最近他们开始通过部署假装赠送资金但实际上包含隐藏陷阱的合约来引诱用户进入陷阱。这种新型合约通常被称为蜜罐合约。

定义 3.蜜罐合约. 蜜罐是一种智能合约, 它假装

将其资金泄露给任意用户, 前提是用户向其发送了额外的资金。但是, 用户提供的资金将被困住, 只有蜜罐创建者能够提取该资金。

区块链上不仅有攻击者部署的恶意蜜罐合约, 还有学者为了捕捉攻击者信息而部署的蜜罐合约。故本小节会分别介绍恶意蜜罐检测模型(蜜罐由攻击者部署)和良性蜜罐部署模型(蜜罐由学者部署)的研究现状。

4.3.1 恶意蜜罐检测模型

蜜罐检测研究难度在于, 一方面, 这是一种新型的非法活动, 可研究的标记数据量过少, 类不平衡问题较庞氏骗局更甚; 另一方面, 面对有漏洞的智能合约, 是真漏洞还是陷阱? 人们往往难以分辨, 若是合法用户部署智能合约出现了失误, 那么就需要及时提醒用户对漏洞进行修复; 若是攻击者的有意为之, 那么就是需要检测和追踪的对象。故需要从智能合约的逻辑链出发, 探寻线索。

如图 8 所示, 恶意蜜罐攻击一般包括三个阶段: (1)攻击者设饵: 攻击者部署看似脆弱的蜜罐(EVM、Solidity 编译器以及 Etherscan 均可以作为蜜罐部署的载体), 以资金的形式放置诱饵。(2)受害者咬饵: 受害者试图通过转入少量资金实现更多收益, 但未能成功。(3)回收资金: 攻击者将诱饵连同受害者注入的资金一起收回。

图 8 恶意蜜罐流程

Figure 8 Malicious honeypot process

对应的, 恶意蜜罐的检测逻辑与上文类似, 数据源包括以太坊交易数据、智能合约数据以及公布的恶意蜜罐。研究方法主要分为两种: 一种先总结恶意蜜罐的行为特征, 然后采用启发式方法正向发现潜在的恶意蜜罐[95-96]; 另一种先提取数据特征, 然后利用机器学习方法实现对恶意蜜罐的检测[97-98]。

Torres 等[95]发现可以在 EVM、Solidity 编译器以及 Etherscan 中寻找到恶意蜜罐的行为语言。

(1) EVM: 主要使用平衡失调技术, 攻击者创建的智能合约中包含一项可将金额转移到任意地址的错做, 但调用条件为金额数需高于现有值, 受害者观察到此情况并输入差额, 却发现缺少执行语句, 导致资金提取失败。

(2) Solidity 编译器: 攻击者通过设置双重变量(区分提取资金变量与输入资金变量)、空字符串(受害者提取资金时出现代码错误)、运行条件(受害者可提取资金远小于注入资金)、数据类型(受害者无法得到正确的提取随机数), 导致受害者无法提取资金。

(3) Etherscan: 攻击者会通过更新隐藏状态、隐藏传输、建立稻草人契约等方式将蜜罐中的资金转移至自身账户中。

如表 13 所示, Torres 等[95]首次对蜜罐智能合约进行系统分析。他们采用符号执行方法, 定义了一种自动检测蜜罐合约的启发式方法, 并从行为、多样性和活动的角度分析了蜜罐合约; 构建的 HONEYBADGER 检测器可以以非常低的误报率有效地检测蜜罐合约。对超过 200 万份智能合约的大规模分析显示, 目前以太坊区块链上部署了 600 多个蜜罐。此外, 对已识别的蜜罐子集的深入分析表明, 已有 240 名用户成为蜜罐的受害者, 攻击者已经获得了超过 90000 美元的利润。在此基础上, 他们创建了一个警示系统[96], 具备自动扫描新部署的蜜罐合约功能, 并在用户将资金发送到蜜罐时提出警告。该系统由一个以太坊钱包 MetaMask 插件和一个后端服务组成——后端服务持续扫描以太坊区块链中的蜜罐, 利用 HONEYBADGER 检测器对蜜罐合约进行识别, 然后将检测结果发送给 MetaMask 插件, 最终反馈给用户。

表 13 恶意蜜罐检测模型

Table 13 Malicious honeypot detection model

时间	作者	检测方法	数据收集	特征提取	蜜罐检测	贡献
2019	Torres 等[95]	启发式方法	以太坊智能合约+交易数据	合约特征+交易特征	符号分析	HONEYBADGER 检测器
2020	Torres 等[96]	启发式方法	以太坊智能合约+交易数据	合约特征+交易特征	符号分析	警示系统
时间	作者	检测方法	数据收集	特征提取	检测模型	F1_score
2019	Camino 等[97]	机器学习	以太坊智能合约+交易数据	源代码特征+交易特征+资金流	XGBoost	--
2020	Chen 等[98]	机器学习	以太坊智能合约	合约特征	LightGBM	0.93

Camino 等[97]使用机器学习方法研究蜜罐合约的检测问题。特征提取是机器学习非常重要的一个环节, Camino 添加了交易聚合特征, 例如交易数量和对应的均值以及其他合约特征, 在合约创建者、合

约、交易发送者和其他参与者之间创建了所有可能的资金流动情况的分区, 然后建立 XGBoost 模型并进行训练。最终成功建立了检测模型, 并发现了两种以前不为人知的新蜜罐合约技术。

由于上述检测模型的特征提取阶段均参考了交易数据, 同庞氏骗局类似, 这种检测模型很难在骗局发生之前提醒用户。Chen 等[98]仅提取合约字节码的特征, 建立基于 N-gram 特征和 LightGBM 的机器学习模型来检测蜜罐合约, 可以在合约部署的那一刻就发出预警。实验结果表明, 具有一元语言(unigram)+二元语言(bigram)特征的模型 F1 值高达 0.93。

4.3.2 良性蜜罐部署模型

与包括但不限于防火墙和入侵检测系统在内的传统方法相比, 蜜罐彻底颠覆了网络防御领域的被动性, 因此受到了网络安全领域的广泛关注[99-105]。部分学者考虑利用蜜罐的欺骗特性吸引攻击者注意。

以太坊上存在这样一类攻击, 攻击者试图寻找存在合约漏洞的以太坊地址, 然后利用漏洞将该地址持有的以太币转移至自身拥有的账户。于是, 学者考虑在以太坊上部署蜜罐收集攻击者的行为信息。主要分为三阶段[106-107]: (1)部署蜜罐: 在以太坊主网中注册一个拥有以太币的真实地址, 用存在漏洞的合约吸引攻击者注意; (2)假响应: 在吸引攻击者发出请求后, 利用后端进行假响应, 实现保护自身的目的; (3)信息收集与分析: 将攻击者发出的请求与调用等信息记录在日志中, 同时搜集以太坊上对应交易信息和地址, 将数据进行统一整理和分析, 得出攻击者的行为特征。以下进行具体分析。

如表 14 所示, Cheng 等[106]对窃取加密货币的攻击行为进行了第一次系统研究。由于以太坊节点上存在未受保护的远程过程调用传送协议(JSON-RPC)端点, 攻击者可以利用这些端点将以太币和其他令牌转移到攻击者控制的帐户。作者通过部署可以捕捉真实攻击的蜜罐实现对攻击者信息的收集, 包括攻击者的 IP 地址、攻击模式和整体利益。在此基础上, Hara 等[107]在 9 个国家都安装了蜜罐, 通过收集恶意通信历史、以太坊网络信息和暗网到达数据包来阐明攻击者的行为和请求倾向, 并建立相应的安全措施。然而, 使用蜜罐进行长期观测会耗费大量运行成本。因此, Chin 等[108]重点关注恶意用户在攻击前进行的预调查, 并使用蜜罐系统对攻击进行观察和分析, 以增强蜜罐的长期有效安装。

表 14 良性蜜罐部署模型

Table 14 Benign honeypot deployment model

时间	作者	蜜罐部署	信息收集	攻击者行为特点	贡献
2019	Cheng 等[106]	以太坊节点(JSON-RPC API 的默认端口)	API 调用信息; 以太坊交易记录	1. 找到具有不安全 HTTP JSON-RPC 端点; 2. 发出 RPC 请求; 3. 窃取以太币。	首次研究
2020	Hara 等[107]	Geth 节点	恶意通信历史; 以太坊网络信息; 暗网到达数据包	1. IP 地址唯一; 2. 获得节点信息; 3. IP 地址隐藏在主网中; 4. 调查的 21 个 IP 地址与试图与暗网通信的 IP 地址匹配。	9 个不同国家和地区的蜜罐和恶意用户
2021	Chin 等[108]	以太坊网络; 假平衡响应; 持有以太币。	HTTP 接收的请求; WebSocket 接收的请求; 通信 IP 地址。	1. 调查潜在攻击目标; 2. 调查节点所属的网络; 3. 调查从节点获得的利益。	恶意用户在攻击前执行的预调查

综上发现目前对以太坊上蜜罐合约的检测研究还很少, 属于新兴研究课题, 具有发展潜力。同时, 对恶意蜜罐的检测以及对攻击者的蜜罐吸引都是对以太坊上非法行为的有力打击, 具有正面的现实意义。故值得学者们对蜜罐合约进行深层次的研究, 研究蜜罐的意义和广泛应用。

5 以太坊非法交易检测技术比较分析

前两节分别对以太坊中非法交易的通用检测技术与特殊检测技术进行梳理, 本节将寻找以太坊上三种特殊非法交易检测方法、通用检测方法与特殊检测方法的异同点, 评价检测方法的“通用性”。

5.1 以太坊网络钓鱼诈骗 VS 庞氏骗局 VS 恶意蜜罐

本节综合对比以太坊上网络钓鱼诈骗、庞氏骗局与恶意蜜罐的行为特点与检测方法, 总结其侧重点与异同点。

5.1.1 交易行为对比

传统场景下, 非法交易的发起者往往通过虚假宣传或武力强制等手段实现对受害者的个人财产侵害, 且只有损害成为既定事实后才会后知后觉, 难以在事前进行预防与发现。而以太坊以去中心化、

匿名化、基于智能合约与账户等特点成为了一个天然的金融交易生态系统, 吸引了非法交易发起者的目光。但由于以太坊属于公有链, 链上数据可以随意查询, 这为非法交易的检测提供了有利途径。

以太坊交易数据包含链上交易数据(硬币交易网络、令牌交易网络)与智能合约代码两类。以下从交易流程角度对比网络钓鱼诈骗、庞氏骗局与恶意蜜罐三种非法交易“迁移”至以太坊上的新特点:

(1) 交易的宣传机制: 网络钓鱼诈骗与庞氏骗局属于“主动攻心”战, 通过借助大众对以太坊这一新兴技术的模糊了解, 使用各种虚假广告吸引用户上当受骗, 此类信息更多体现在日常社交网络上, 并未在区块链上有所体现; 恶意蜜罐属于“被动设陷”, 通过发布具有“伪漏洞”的智能合约吸引以太坊用户注意力。

(2) 交易的运转机制: 网络钓鱼诈骗往往是一次性的, 用户往特定的账户注入资金即完成; 庞氏骗局与恶意蜜罐需要依靠智能合约的强制执行进行运转, 故发起者在前期需要将运转机制转化为智能合约保存在区块链上, 这表示通过分析智能合约代码数据, 学者有可能在交易发生前实现检测和预警。

(3) 交易的收网机制: 网络钓鱼诈骗、庞氏骗局与恶意蜜罐的最终目的都是骗取以太坊用户的资金(以太币), 当用户将资金转入特定账户即视为“完成”交易, 故链上交易数据记录着发起者的账户地址、获利金额数以及时间戳等信息, 是学者实现检测的重要信息来源。

综上所述, 网络钓鱼诈骗的痕迹主要从链上交易数据中寻找; 庞氏骗局与恶意蜜罐的运转机制隐藏在智能合约中且通过转账实现交易, 故智能合约代码和链上交易数据中均包含有效信息。

5.1.2 检测方法对比

首先对检测环节进行分析, 非法交易检测一般包含数据收集、特征提取、异常检测三个环节。

(1) 数据收集: 因为非法交易检测属于二分类问题, 故带标记数据有助于训练模型参数, 可从Etherscan 等网站中寻找已公布的恶意账户。

(2) 特征提取: 网络钓鱼诈骗的检测无法借助智能合约代码数据, 且链上交易数据呈现高维异构特点, 需要借助图嵌入、深度学习等方法挖掘深层次信息; 庞氏骗局和恶意蜜罐的检测可以从智能合约代码和链上交易数据中提取特征, 而且在庞氏骗局检测研究中, 发现智能合约特征比链上交易特征更具有代表性。

(3) 异常检测: 三种非法交易一般都是利用机器学习的分类或聚类算法实现有效检测, 其中 LR、LightGBM 等方法精度较高。

其次就检测难度而言, 网络钓鱼诈骗>恶意蜜罐>庞氏骗局。因为网络钓鱼检测的隐蔽性较高, 且仅存在链上交易数据可供使用, 无法使用启发式方法提高检测效率。在后两种交易检测中, 恶意蜜罐智能合约的部署具有迷惑性, 识别漏洞的真伪性有一定技术难度; 而庞氏骗局的运转机制具有“发展下线——返佣”内核, 检测模型的精度最高。

最后讨论检测方法的“迁移性”。通过观察发现非法交易检测重在提取具有代表性的特征。针对智能合约代码, 庞氏骗局的检测思路可以借鉴到恶意蜜罐智能合约检测中; 针对链上交易数据, 网络钓鱼诈骗中基于图嵌入和深度学习的提取方法可以给其余两类非法交易检测启示。

5.2 以太坊通用检测 VS 特殊检测

在以太坊上, 非法交易的通用检测与特殊检测区别主要体现在以下三个方面:

(1)数据收集方面, 通用检测针对所有非法交易, 统一收集被标记为非法交易的账户地址和交易信息; 特殊检测则关注到具体的某一类非法交易(例如以太坊网络钓鱼诈骗、庞氏骗局等), 从网站上收集带标签数据进行研究。

(2)特征提取方面, 通用检测重在分析交易网络中的结构信息与时间信息, 从中提取具有代表性的特征; 特殊检测会分析智能合约代码潜藏的语义信息, 将合约特征与交易特征相结合, 提高检测精度(例如检测以太坊上庞氏骗局的全特征模型)。

(3)检测技术方面, 通用检测以基于机器学习的算法为主, 采用无监督算法发现未知非法交易、采用有监督算法将现有交易分类为非法与合法交易; 特殊检测则占据特定交易优势, 可以使用启发式方法总结交易的行为模式及其在智能合约与交易流程中的体现, 有助于挖掘智能合约语义信息。

综上所述, 通用检测方法与特殊检测方法各有侧重点。前者适合发掘新的非法交易模式, 后者适合对特定的非法交易模式“一网打尽”。且两者的检测方法大致相同, 可以相互借鉴参考。

6 区块链间非法交易检测技术比较分析

在过去的十年内, 区块链技术由于其去中心化、匿名化等特点引起了行业和学界的广泛关注, 诞生了包括比特币、以太坊在内的诸多区块链平台。本节将以比特币、以太坊、Rootstock、EOS、Fabric、Corda 为例进行区块链上非法交易检测方法的比较。

首先研究智能合约功能的影响, 对比比特币与以太坊的非法交易检测方法。然后专注于拥有智能合约功能的区块链平台, 比较不同数据结构与运行机制下非法交易检测方法的迁移性。

6.1 比特币 VS 以太坊

首先比较比特币和以太坊交易机制的异同点。相同之处在于链上交易为公开数据集, 交易信息包括交易发送端地址、接收端地址、交易金额数、时间戳、随机数等。不同之处在于交易主体, 比特币支持一个用户同时拥有多个地址, 每个地址携带比特币进行交易, 形成的交易网络为 UXTO 交易网络; 以太坊支持一个用户使用固定地址进行长期交易, 且利用智能合约功能发展出各种 DAPP 应用, 扩大金融业务范围, 生成的交易网络包括硬币交易网络与令牌交易网络两类。综合上述信息, 比特币的难点在于实体识别, 即将属于同一用户的地址聚合起来进行去匿名化; 以太坊的难点在于识破伪装, 即在花样繁多的智能合约与交易行为中发现非法交易的内在联系。如表 15 所示。

表 15 比特币与以太坊上非法交易检测特点比较

Table 15 Comparison of detection characteristics of illegal transactions on Bitcoin and Ethereum

区块链	数据收集	特征提取	异常检测
比特币	链上交易数据	交易特征	
以太坊	链上交易数据	交易特征	各类机器学习算法
	智能合约代码	字节码特征	

Shin 等[37]比较了比特币和以太坊上地址聚类方法的区别。发现在比特币中, 许多关于钱包地址聚类的研究都是基于 txs 中存储的数据进行的; 然而该数据无法为以太坊地址聚类提供帮助。因为在比特币, 由相同私钥管理的钱包地址可以列在 tx 的 vin 中; 与比特币相反, 在以太坊中一个 tx 中只能写入一个发送地址。这表明无法直接将比特币中的检测模型照搬到以太坊中。

但另一方面, 比特币与以太坊的交易数据均以交易网络的形式存在, 故关于链上交易特征的提取思路可以相互借鉴, 尤其是同一种非法交易在不同区块链上的检测(例如庞氏骗局)。Bartoletti 等[109]在研究比特币庞氏骗局的检测模型时, 提取地址集群中的特征, 采用 RF 的方法进行检测, 实现了高达 0.997 的精度。同样的, RF 同样被 Jung 等[86]用于检测以太坊上的庞氏骗局, 检测精度达到 0.93。

6.2 以太坊 VS 其他智能合约平台

由于智能合约具有人为编辑、不可更改和强制执行特点, 受到了区块链开发者的青睐。

表 16 从类型、数据结构、智能合约语言、运行环境与访问权限等方面对以太坊、Rootstock、EOS、Fabric、Corda 等区块链平台进行比较分析, 讨论本文整理的检测方法对其他智能合约平台的适用性。

表 16 智能合约平台特点比较

Table 16 Feature comparison of smart contract platform

区块链	类型	访问权限	数据结构	智能合约语言	运行环境
以太坊	公链	可访问	账户	Solidity, Serpent, LLL, Mutan	EVM
Rootstock	公链	可访问	账户	Solidity	VM
EOS	公链	可访问	账户	C++	WebAssembly
Fabric	私链或联盟链	需授权	键值对	Java, Golang	Docker
Corda	私链	需授权	交易	Java, Kotlin6	JVM

(1)类型与访问权限: 现如今存在公链、私链和联盟链三种类型的区块链平台, 其中公链向全世界开放, 例如以太坊, 任何人均可加入且查看链上交易信息; 私链属于个人(或公司)所有, 例如 Fabric, 用户必须获得授权才能加入, 多用于企业内部管理; 联盟链是指由多个机构共同参与管理的区块链, 一般来说适合于机构间的交易、结算或清算等 B2B 场景。从中可看出私链和联盟链属于中心化或半中心化结构, 参与者身份透明, 制定完备的智能合约即可满足交易的安全执行, 不存在非法交易问题; 公链属于去中心化结构, 非法交易的检测有其应用价值, 故可以考虑将本文的检测方法应用至公链平台。

(2) 数据结构: 从数据角度来看, 以太坊、Rootstock、EOS 中的交易主体以账户的形式存在。其中 Rootstock 作为挂钩于比特币网络的智能合约平台, 利用侧链实现新功能, 扩展比特币应用领域; EOS 允许所有账户被人类可读的名称引用, 而不是像以太坊那样为代码执行 EVM。因为基于账户的一致性, 研究非法账户的检测思路可以共享。

(3)智能合约语言: 通过对智能合约的语义分析, 有助于发现非法交易的逻辑及其字节码表达(例如以太坊上庞氏骗局智能合约特征)。以太坊支持专为以太坊设计的 Solidity、Serpent 和 Mutan。为了与以太坊兼容, Rootstock 采用 Solidity 作为合约语言, 而 EOS 目前仅支持 C++。故以太坊上关于非法交易代码特征对于 Rootstock 同样适用; 对于 EOS 等语言不一致的平台而言, 可以根据语义逻辑对应到 C++等语言中, 寻找类似的表达结构。

(4)运行环境: 运行环境同样会对非法交易的检测造成影响[95]。以太坊中的合约在 EVM 中执行; Fabric 在 Docker 之上运行智能合约, 从而减少了开销, 同时减少了应用程序的隔离; EOS 选择使用 WebAssembly 来支持更多智能合约语言, 具有更高的自由度, 但同样会导致非法交易表现形式的多样化, 检测难度进一步加大。不同的运行环境表示以太坊上对 EVM 的观察结果无法直接平移到 Rootstock、EOS 等平台中。

综上所述, 对于同样使用智能合约功能的其他区块链平台, 以太坊上非法交易检测方法具有一定的参考价值。但同时需要思考平台的自身特点, 例如使用的智能合约语言和数据结构等存在非一致性, 在迁移过程中强调检测方法的逻辑思维以及语言表示。

7 趋势和挑战

据调查, 目前以太坊已经成为各种非法活动的“天堂”, 其中网络钓鱼诈骗、庞氏骗局、蜜罐骗局是主要非法行为。很多研究文章通过机器学习方法建立检测模型, 旨在成功检测诈骗交易并进行追踪和预警。主要研究重点在于数据集的实时性和极端不平衡性、特征代表性和模型时效性。据此, 本章提出了未来工作的 5 个研究方向。

(1)以太坊数据的多维性和实时性研究。以太坊不仅包含链上交易数据, 还包含智能合约数据, 呈现出数据复杂化、多维化特点, 且以太坊生态系统的本质使得衍生出的网络钓鱼诈骗、庞氏骗局等表现形式多样化; 另一方面以太坊上持续产生交易, 即交易网络一直处于更新状态, 传统的手动收集数据方法已不再适用, 故需要尝试构建实时收集多维数据, 研究数据动态演变趋势的非法交易检测模型。

(2)类的极端不平衡问题研究。以太坊交易量呈爆发式增长, 这其中大部分都是正常交易, 服务于日常生活, 非法交易仅是其中非常小的一部分, 往往很难发现, 而且还有很多新的非法交易形式隐藏其中。现有的已标记的非法交易数据集与正常交易数据集存在数据量极端不平衡问题, 需要寻找合适的采样方法提高模型的分类效果。现如今, 机器学习算法不断在发展, 结合拓扑知识、概率论知识、图网络知识, 会衍生出更多的新的更高效的机器学习模型, 值得关注和研究。

(3)0-day 模型与全特征模型研究。以太坊的数据来源主要有链上交易数据与智能合约代码数据两种, 根据数据来源可以将检测模型分为两类: 0-day 模型与全特征模型。其中, 前者仅包含智能合约代码数据, 后者包含交易数据和代码数据。由于以太坊上的交易一旦开始, 就会自动运行, 无法因为某一方的意愿而终止, 所以事前预警有其必要性。而智能合约代码属于交易前静态数据, 可以提取有效特征进行非法交易检测。故建立依靠智能合约代码数据的 0-day 模型有助于事前预警, 实现零损失。

另一方面, 对交易数据的研究亦有其独特价值。交易数据与智能合约代码数据具有关联性, 能够提取出更具有代表性的特征; 而智能合约一旦部署就会长期存在, 分析交易数据有助于发现正在进行或已完成的非法交易, 使得未参与用户避开诈骗陷阱、已参与用户及时止损(例如庞氏骗局); 进一步地, 从交易数据的动态演变中能观察出以太坊上非法交易套路的“提高”和“创新”, 故借助交易数据探究链上已有的非法交易过程、及时追踪实体身份、避免被“新招术”上当受骗是非常有价值的。

(4)基于深度学习的检测模型研究。以太坊交易网络主要分为硬币交易网络与令牌交易网络, 每笔交易对应唯一的发送方和接收方, 以太坊交易网络复杂且数据维度高, 研究相当来说较为困难。但是, 因为有着类似网络结构和数据特征, 使得基于深度学习的一些社交网络的研究方法可以顺延拓展到以太坊上。图嵌入算法可以有效降低交易网络的数据维度, 将大规模稀疏的高维节点向量转化为密集的低维节点向量, 实现对多维数据的精准研究, 在网络钓鱼诈骗交易数据特征提取中表现出良好的性能。其他诸如图的级联提取、图分类等方式也可以较好地提取出网络和数据特征, 未来, 基于深度学习的检测模型将有望实现对非法交易的精准探测。

(5)以太坊非法检测软件的落地和共享研究。现如今已有不少关于以太坊上非法交易的检测模型, 但是几乎没有真正应用到以太坊平台上。如何将训练好的检测模型真正应用到以太坊的交易监管, 真正服务于以太坊用户, 是需要思考的方向; 另一方面, 已经有一些检测算法和软件出现在大众视野, 例如 Etherescan 团队推出以太坊“非法活动”监控。但是由于底层运行代码属商业机密, 缺少监管部门的协调和统一管理, 用户需要同时运行多个检测软件才能保证自己的正常交易, 这也是未来区块链等基建工程提供社会服务时需要思考的管理问题。

8 结论

本文综合调查了以太坊上非法交易和账户的检测模型。主要从两个角度展开: 一方面是从以太坊交

易数据角度出发, 分析时空维度下的交易行为特征, 实现对异常交易行为检测的模型分析, 该类方法具有通用性; 另一方面是针对特定的非法交易类型(主要包括网络钓鱼诈骗、庞氏骗局和蜜罐合约等), 总结研究思路、模型建立及评估效果。最后, 对目前以太坊以及区块链间非法检测研究的现状进行比较分析, 并展望了未来研究方向以及挑战。

参考文献

[1] Squarepants S. Bitcoin: A Peer-to-Peer Electronic Cash System[J]. *SSRN Electronic Journal*, 2008: 21260-21268.

[2] Wood G. Ethereum: A secure decentralised generalised transaction ledger[J]. *Ethereum project yellow paper*, 2014, 151(2014): 1-32.

[3] Hacking Distributed. 2016. Analysis of the DAO exploit, http://hackingdistributed.com/2016/06/18/analysis-of-the-dao-exploit/.

[4] Petrov, Another parity wallet hack explained, Nov 2017, https://medium.com/@Pr0Ger/anotherparity-wallet-hack-explained-847ca46a2e1c.

[5] EtherScamDB, Etherscamdb. [Online]. Available: https://blog.chainalysis.com/reports/the\-rise-of-cybercrime-on-ethereum.

[6] Householder A D, Chrabaszcz J, Novelly T, et al. Historical Analysis of Exploit Availability Timelines[C]. *The 13th USENIX Conference on Cyber Security Experimentation and Test*, 2020: 6.

[7] Gao Z P, Jiang L X, Xia X, et al. Checking Smart Contracts with Structural Code Embedding[J]. *IEEE Transactions on Software Engineering*, 2021, 47(12): 2874-2891.

[8] Wu L, Wu S, Zhou Y, et al. Ethscope: A transaction-centric security analytics framework to detect malicious smart contracts on Ethereum[J]. arXiv preprint arXiv: 2005.08278, 2020.

[9] Signorini M, Pontecorvi M, Kanoun W, et al. BAD: Blockchain Anomaly Detection[EB/OL]. 2018: 1807.03833. http://arxiv.org/abs/1807.03833v3.

[10] Sai K, Tipper D. Disincentivizing Double Spend Attacks across Interoperable Blockchains[C]. *2019 First IEEE International Conference on Trust*, *Privacy and Security in Intelligent Systems and Applications*, 2019: 36-45.

[11] Chen T, Li X, Wang Y, et al. An adaptive gas cost mechanism for Ethereum to defend against underpriced dos attacks[C]. *International conference on information security practice and experience*, 2017: 3-24.

[12] Ao X, Liu Y, Qin Z D, et al. Temporal High-Order Proximity Aware Behavior Analysis on Ethereum[J]. *World Wide Web*, 2021, 24(5): 1565-1585.

[13] Chen T, Zhang Y F, Li Z H, et al. TokenScope: Automatically Detecting Inconsistent Behaviors of Cryptocurrency Tokens in Ethereum[C]. *The 2019 ACM SIGSAC Conference on Computer and Communications Security*, 2019: 1503-1520.

[14] Ibrahim R F, Mohammad Elian A, Ababneh M. Illicit Account Detection in the Ethereum Blockchain Using Machine Learning[C]. *2021 International Conference on Information Technology*, 2021: 488-493.

[15] Farrugia S, Ellul J, Azzopardi G. Detection of Illicit Accounts over the Ethereum Blockchain[J]. *Expert Systems with Applications*, 2020, 150: 113318.

[16] Liu X, Tang Z Y, Li P, et al. A Graph Learning Based Approach for Identity Inference in DApp Platform Blockchain[J]. *IEEE Transactions on Emerging Topics in Computing*, 2022, 10(1): 438-449.

[17] Shin M. Scalable Anomaly Detection Method for Blockchain Transactions Using GPU[C]. *2019 20th International Conference on Parallel and Distributed Computing*, *Applications and Technologies*, 2019: 160-165.

[18] 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.

[19] O' Kane E. Detecting patterns in the Ethereum transactional data using unsupervised learning[D]. Master' s thesis, University of Dublin, Trinity College, Dublin, Ireland, 2018.

[20] Praitheeshan P, Pan L, Yu J S, et al. Security Analysis Methods on Ethereum Smart Contract Vulnerabilities: A Survey[EB/OL]. 2019: 1908.08605. http://arxiv.org/abs/1908.08605v3.

[21] Samreen N F, Alalfi M H. SmartScan: An Approach to Detect Denial of Service Vulnerability in Ethereum Smart Contracts[C]. *2021 IEEE/ACM 4th International Workshop on Emerging Trends in Software Engineering for Blockchain*, 2021: 17-26.

[22] Liu C, Liu H, Cao Z, et al. ReGuard: Finding Reentrancy Bugs in Smart Contracts[C]. *2018 IEEE/ACM 40th International Conference on Software Engineering*: *Companion*, 2018: 65-68.

[23] Huang Y F, Bian Y Y, Li R P, et al. Smart Contract Security: A Software Lifecycle Perspective[J]. *IEEE Access*, 2019, 7: 150184-150202.

[24] Sayeed S, Marco-Gisbert H, Caira T. Smart Contract: Attacks and Protections[J]. *IEEE Access*, 2020, 8: 24416-24427.

[25] Grech N, Kong M, Jurisevic A, et al. MadMax: Surviving Out-of-Gas Conditions in Ethereum Smart Contracts[J]. *Proceedings of the ACM on Programming Languages*, 2018, 2(OOPSLA): 116.

[26] Ul Hassan M, Rehmani M H, Chen J J. Anomaly Detection in Blockchain Networks: A Comprehensive Survey[J]. *IEEE Communications Surveys & Tutorials*, 2023, 25(1): 289-318.

[27] Schmee J. Outliers in Statistical Data (2nd Ed.)[J]. *Technometrics*, 1986, 28: 89-90.

[28] Ted J, Ivy K, Raymond Ng. Fast computation of 2-dimensional depth contours[C]. *The Fourth International Conference on Knowledge Discovery and Data Mining*, 1998: 224-228.

[29] Kroger, P. Outlier Detection Techniques[C]. *Acm Sigkdd International Conference on Knowledge Discovery & Data Mining ACM*,

2010.

[30] Breunig M, Kriegel H, Raymond T. Ng, et al. OPTICS-OF: Identifying Local Outliers[C]. *The Third European Conference on Principles of Data Mining and Knowledge Discovery*, 1999: 262-270.

[31] Akoglu L, Tong H H, Koutra D. Graph Based Anomaly Detection and Description: A Survey[J]. *Data Mining and Knowledge Discovery*, 2015, 29(3): 626-688.

[32] Ofori-Boateng D, Dominguez I S, Akcora C, et al. Topological Anomaly Detection in Dynamic Multilayer Blockchain Networks[C]. *Machine Learning and Knowledge Discovery in Databases*. *Research Track*: *European Conference*, 2021: 788-804.

[33] Buterin, V., et al. A next-generation smart contract and decentralized application platform[J]. *White Paper*, 2014: 3(37).

[34] Akcora C G, Gel Y R, Kantarcioglu M. Blockchain Networks: Data Structures of Bitcoin, Monero, Zcash, Ethereum, Ripple, and Iota[J]. *WIREs Data Mining and Knowledge Discovery*, 2022, 12(1): e1436.

[35] Klusman R., Dijkhuizen T. Deanonymisation in Ethereum Using Existing Methods for Bitcoin. https://rp.os3.nl/2017-2018/p61/ report.pdf. Feb, 2018.

[36] Victor F. Address Clustering Heuristics for Ethereum[C]. *International Conference on Financial Cryptography and Data Security*, 2020: 617-633.

[37] Shin H Y, Essaid M, Park S, et al. A Survey on Public Blockchain-Based Networks: Structural Differences and Address Clustering Methods[C]. *2021 22nd Asia-Pacific Network Operations and Management Symposium*, 2021: 57-60.

[38] Wu Z, Liu J, Wu J, et al. Transaction Tracking on Blockchain Trading Systems using Personalized PageRank[J]. 2022: ArXiv Preprint. ArXiv: 2201.05757.

[39] Dyson S F, Buchanan W J, Bell L. Scenario-Based Creation and Digital Investigation of Ethereum ERC20 Tokens[J]. *Forensic Science International*: *Digital Investigation*, 2020, 32: 200894.

[40] Lin D, Wu J J, Xuan Q, et al. Ethereum Transaction Tracking: Inferring Evolution of Transaction Networks via Link Prediction[J]. *Physica A*: *Statistical Mechanics and Its Applications*, 2022, 600: 127504.

[41] Bang J, Choi M J. Design and Implementation of Storage System for Real-Time Blockchain Network Monitoring System[C]. *2019 20th Asia-Pacific Network Operations and Management Symposium*, 2019: 1-4.

[42] Lee C, Kim H, Maharjan S, et al. Blockchain Explorer Based on RPC-Based Monitoring System[C]. *2019 IEEE International Conference on Blockchain and Cryptocurrency*, 2019: 117-119.

[43] Zhou J J, Hu C K, Chi J L, et al. Behavior-Aware Account De-Anonymization on Ethereum Interaction Graph[J]. *IEEE Transactions on Information Forensics and Security*, 2022, 17: 3433-3448.

[44] 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.

[45] Zhao J P. Research on the Investigation into Criminal Cases of Network Pyramid Selling by Virtual Currency[D]. Beijing: Chinese People's Public Security University, 2018.
(赵军鹏. 利用虚拟货币网络传销犯罪案件侦查研究[D]. 北京: 中国人民公安大学, 2018.)

[46] Sun H Y, Ruan N, Liu H Q. Ethereum Analysis via Node Clustering[C]. *International Conference on Network and System Security*, 2019: 114-129.

[47] Poursafaei F, Hamad G B, Zilic Z. Detecting Malicious Ethereum Entities via Application of Machine Learning Classification[C]. *2020 2nd Conference on Blockchain Research & Applications for Innovative Networks and Services*, 2020: 120-127.

[48] Agarwal R, Barve S, Shukla S K. Detecting Malicious Accounts in Permissionless Blockchains Using Temporal Graph Properties[J]. *Applied Network Science*, 2021, 6(1): 9.

[49] Agarwal R, Thapliyal T, Shukla S K. Detecting malicious accounts showing adversarial behavior in permissionless blockchains[J]. arXiv preprint arXiv:2101.11915, 2021.

[50] Hu H, Xu Y. SCSGuard: Deep Scam Detection for Ethereum Smart Contracts[J]. arXiv preprint arXiv:2105.10426, 2021.

[51] Baek H, Oh J, Kim C Y, et al. A Model for Detecting Cryptocurrency Transactions with Discernible Purpose[C]. *2019 Eleventh International Conference on Ubiquitous and Future Networks*, 2019: 713-717.

[52] Poursafaei F, Rabbany R, Zilic Z. SigTran: Signature Vectors for Detecting Illicit Activities in Blockchain Transaction Networks[C]. *Pacific-Asia Conference on Knowledge Discovery and Data Mining*, 2021: 27-39.

[53] Sachan R K, Agarwal R, Shukla S K. Identifying malicious accounts in Blockchains using Domain Names and associated temporal properties[J]. arXiv preprint arXiv:2106.13420, 2021.

[54] Olson R S, Moore J H. TPOT: A Tree-Based Pipeline Optimization Tool for Automating Machine Learning[M]. Automated Machine Learning, 2019: 151-160.

[55] Khonji M, Iraqi Y, Jones A. Phishing Detection: A Literature Survey[J]. *IEEE Communications Surveys & Tutorials*, 2013, 15(4): 2091-2121.

[56] Dou Z C, Khalil I, Khreishah A, et al. Systematization of Knowledge (SoK): A Systematic Review of Software-Based Web Phishing Detection[J]. *IEEE Communications Surveys & Tutorials*, 2017, 19(4): 2797-2819.

[57] Chen W, Guo X, Chen Z, et al. Phishing Scam Detection on Ethereum: Towards Financial Security for Blockchain Ecosystem[C]. *IJCAI*. 2020: 4506-4512.

[58] Alam M N, Sarma D, Lima F F, et al. Phishing Attacks Detection

Using Machine Learning Approach[C]. *2020 Third International Conference on Smart Systems and Inventive Technology*, 2020: 1173-1179.

[59] Zhang D K, Yin J, Zhu X Q, et al. Homophily, Structure, and Content Augmented Network Representation Learning[C]. *2016 IEEE 16th International Conference on Data Mining*, 2016: 609-618.

[60] Yang C, Liu Z, Zhao D, Sun M, Chang E. Y, Network representation learning with rich text information[C]. *IJCAI*, 2015: 2111-2117.

[61] Backstrom L, Leskovec J. Supervised Random Walks: Predicting and Recommending Links in Social Networks[C]. *The fourth ACM international conference on Web search and data mining*, 2011: 635-644.

[62] Chau D H, Nachenberg C, Wilhelm J, et al. Polonium: Tera-Scale Graph Mining and Inference for Malware Detection[C]. *The 2011 SIAM International Conference on Data Mining*, 2011: 131-142.

[63] Duvenaud D, Maclaurin D, Aguilera-Iparraguirre J, et al. Convolutional Networks on Graphs for Learning Molecular Fingerprints[C]. *The 28th International Conference on Neural Information Processing Systems - Volume 2*, 2015: 2224-2232.

[64] Cai H Y, Zheng V W, Chang K C C. A Comprehensive Survey of Graph Embedding: Problems, Techniques, and Applications[J]. *IEEE Transactions on Knowledge and Data Engineering*, 2018, 30(9): 1616-1637.

[65] Goyal P, Ferrara E. Graph Embedding Techniques, Applications, and Performance: A Survey[J]. *Knowledge-Based Systems*, 2018, 151: 78-94.

[66] Adhikari B, Zhang Y, Ramakrishnan N, et al. Distributed Representations of Subgraphs[C]. *2017 IEEE International Conference on Data Mining Workshops*, 2017: 111-117.

[67] Cao S S, Lu W, Xu Q K. Deep Neural Networks for Learning Graph Representations[C]. *The Thirtieth AAAI Conference on Artificial Intelligence*, 2016: 1145-1152.

[68] Shervashidze N, Schweitzer P, Van Leeuwen E J, et al. Weisfeiler-Lehman Graph Kernels[J]. *Journal of Machine Learning Research*, 2011, 12: 2539-2561.

[69] Yuan Q, Huang B Y, Zhang J, et al. Detecting Phishing Scams on Ethereum Based on Transaction Records[C]. *2020 IEEE International Symposium on Circuits and Systems*, 2020: 1-5.

[70] Yuan Z H, Yuan Q, Wu J J. Phishing Detection on Ethereum via Learning Representation of Transaction Subgraphs[C]. *International Conference on Blockchain and Trustworthy Systems*, 2020: 178-191.

[71] Wang J H, Chen P T, Yu S Q, et al. TSGN: Transaction Subgraph Networks for Identifying Ethereum Phishing Accounts[C]. *International Conference on Blockchain and Trustworthy Systems*, 2021: 187-200.

[72] 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.

[73] Lin D, Wu J J, Yuan Q, et al. T-EDGE: Temporal WEighted MultiDiGraph Embedding for Ethereum Transaction Network Analysis[J]. *Frontiers in Physics*, 2020, 8: 204.

[74] Xie Y Y, Zhou J J, Wang J H, et al. Understanding Ethereum Transactions via Network Approach[M]. Graph Data Mining. Singapore: Springer, 2021: 155-176.

[75] Chen L, Peng J Y, Liu Y, et al. Phishing Scams Detection in Ethereum Transaction Network[J]. *ACM Transactions on Internet Technology*, 2020, 21(1): 10.

[76] Li S C, Xu F Y, Wang R C, et al. Self-Supervised Incremental Deep Graph Learning for Ethereum Phishing Scam Detection[EB/OL]. 2021: 2106.10176. http://arxiv.org/abs/2106. 10176v1.

[77] Zhang D J, Chen J Y, Lu X S. Blockchain Phishing Scam Detection via Multi-Channel Graph Classification[C]. *International Conference on Blockchain and Trustworthy Systems*, 2021: 241-256.

[78] Kanezashi H, Suzumura T, Liu X, et al. Ethereum Fraud Detection with Heterogeneous Graph Neural Networks[EB/OL]. 2022: 2203.12363. http://arxiv.org/abs/2203.12363v3.

[79] Chen H L, Wang Z B, Xia F, et al. Efficiently and Completely Identifying Missing Key Tags for Anonymous RFID Systems[J]. *IEEE Internet of Things Journal*, 2018, 5(4): 2915-2926.

[80] Chen J Y, Xiong H Y, Zhang D J, et al. TEGDetector: A Phishing Detector that Knows Evolving Transaction Behaviors[EB/OL]. 2021: 2111.15446. http://arxiv.org/abs/2111.15446v1.

[81] Vasek M, Moore T. Analyzing the Bitcoin Ponzi Scheme Ecosystem[C]. *International Conference on Financial Cryptography and Data Security*, 2019: 101-112.

[82] Chen W M, Li X R, Sui Y T, et al. SADPonzi: Detecting and Characterizing Ponzi Schemes in Ethereum Smart Contracts[J]. *Proceedings of the ACM on Measurement and Analysis of Computing Systems*, 2021, 5(2): 26.

[83] Bartoletti M, Carta S, Cimoli T, et al. Dissecting Ponzi Schemes on Ethereum: Identification, Analysis, and Impact[J]. *Future Generation Computer Systems*, 2020, 102: 259-277.

[84] Chen W L, Zheng Z B, Cui J H, et al. Detecting Ponzi Schemes on Ethereum: Towards Healthier Blockchain Technology[C]. *The 2018 World Wide Web Conference*, 2018: 1409-1418.

[85] Chen W L, Zheng Z B, Ngai E C H, et al. Exploiting Blockchain Data to Detect Smart Ponzi Schemes on Ethereum[J]. *IEEE Access*, 2019, 7: 37575-37586.

[86] Jung E, Le Tilly M, Gehani A, et al. Data Mining-Based Ethereum Fraud Detection[C]. *2019 IEEE International Conference on Blockchain*, 2019: 266-273.

[87] Wang L, Cheng H, Zheng Z B, et al. Ponzi Scheme Detection via

Oversampling-Based Long Short-Term Memory for Smart Contracts[J]. *Knowledge-Based Systems*, 2021, 228: 107312.

[88] Zhang Y M, Yu W Q, Li Z Y, et al. Detecting Ethereum Ponzi Schemes Based on Improved LightGBM Algorithm[J]. *IEEE Transactions on Computational Social Systems*, 2022, 9(2): 624-637.

[89] Liang Y Z, Wu W J, Lei K, et al. Data-Driven Smart Ponzi Scheme Detection[EB/OL]. 2021: 2108.09305. http://arxiv.org/abs/2108.09305v1.

[90] Yu S Q, Jin J, Xie Y Y, et al. Ponzi Scheme Detection in Ethereum Transaction Network[C]. *International Conference on Blockchain and Trustworthy Systems*, 2021: 175-186.

[91] Fan S H, Fu S J, Xu H R, et al. Expose Your Mask: Smart Ponzi Schemes Detection on Blockchain[C]. *2020 International Joint Conference on Neural Networks*, 2020: 1-7.

[92] Sun W S, Xu G Y, Yang Z J, et al. Early Detection of Smart Ponzi Scheme Contracts Based on Behavior Forest Similarity[C]. *2020 IEEE 20th International Conference on Software Quality, Reliability and Security*, 2020: 297-309.

[93] Bian L Y, Zhang L L, Zhao K, et al. Image-Based Scam Detection Method Using an Attention Capsule Network[J]. *IEEE Access*, 2021, 9: 33654-33665.

[94] Kipf T N, Welling M. Semi-Supervised Classification with Graph Convolutional Networks[EB/OL]. 2016: 1609.02907. http://arxiv.org/abs/1609.02907v4.

[95] orres C F, Steichen M, State R. The Art of the Scam: Demystifying Honeypots in Ethereum Smart Contracts[C]. *The 28th USENIX Conference on Security Symposium*, 2019: 1591-1607.

[96] Torres C F, Baden M, State R. Towards Usable Protection Against Honeypots[C]. *2020 IEEE International Conference on Blockchain and Cryptocurrency*, 2020: 1-2.

[97] Camino R, Torres C F, Baden M, et al. A Data Science Approach for Detecting Honeypots in Ethereum[C]. *2020 IEEE International Conference on Blockchain and Cryptocurrency*, 2020: 1-9.

[98] Chen W L, Guo X F, Chen Z G, et al. Honeypot Contract Risk Warning on Ethereum Smart Contracts[C]. *2020 IEEE International Conference on Joint Cloud Computing*, 2020: 1-8.

[99] Qassrawi M T, Zhang H L. Client Honeypots: Approaches and Challenges[C]. *4th International Conference on New Trends in Information Science and Service Science*, 2010: 19-25.

[100] Qassrawi M T, Zhang H L. Deception Methodology in Virtual Honeypots[C]. *2010 Second International Conference on Networks Security, Wireless Communications and Trusted Computing*, 2010: 462-467.

[101] Lavrov D, Blanchet V, Pang S N, et al. COR-Honeypot: Copy-on-Risk, Virtual Machine as Honeypot in the Cloud[C]. *2016 IEEE 9th International Conference on Cloud Computing*, 2016: 908-912.

[102] Park B, Dang S P, Noh S, et al. Dynamic Virtual Network Honeypot[C]. *2019 International Conference on Information and Communication Technology Convergence*, 2019: 375-377.

[103] Dodson M, Beresford A R, Vingaard M. Using Global Honeypot Networks to Detect Targeted ICS Attacks[C]. *2020 12th International Conference on Cyber Conflict*, 2020: 275-291.

[104] Mudgal A, Bhatia S. A Step towards Improvement in Classical Honeypot Security System[C]. *2022 International Conference on Machine Learning, Big Data, Cloud and Parallel Computing*, 2022: 720-725.

[105] Song Y B, Zhu X Y, Hong Y L, et al. A Mobile Communication Honeypot Observing System[C]. *2012 Fourth International Conference on Multimedia Information Networking and Security*, 2012: 861-865.

[106] Cheng Z, Hou X R, Li R H, et al. Towards a First Step to Understand the Cryptocurrency Stealing Attack on Ethereum [EB/OL]. 2019: 1904.01981. http://arxiv.org/abs/1904.01981v2.

[107] Hara K, Sato T, Imamura M, et al. Profiling of Malicious Users Using Simple Honeypots on the Ethereum Blockchain Network[C]. *2020 IEEE International Conference on Blockchain and Cryptocurrency*, 2020: 1-3.

[108] Chin K, Omote K. Analysis of Attack Activities for Honeypots Installation in Ethereum Network[C]. *2021 IEEE International Conference on Blockchain*, 2021: 440-447.

[109] Bartoletti M, Pes B, Serusi S. Data Mining for Detecting Bitcoin Ponzi Schemes[C]. *2018 Crypto Valley Conference on Blockchain Technology*, 2018: 75-84.

李梦 于 2019 年在合肥工业大学应用数学专业获得硕士学位。现任江苏警官学院讲师。研究领域为区块链、网络安全方向。CCF 专业会员。Email: limeng@jspi.cn

梁广俊 于 2018 年在南京邮电大学通信与信息系统专业获得博士学位。现在江苏警警官学院担任副教授。研究领域为无线通信网络中的资源分配与优化、中继通信、网络安全。CCF 专业会员。Email: lianggjun@126.com

印杰 于 2008 年获得南京理工大学软件工程硕士学位。江苏警官学院计算机信息与网络安全系, 高级工程师。江苏省电子数据取证与分析工程研究中心、江苏省公安厅数字取证重点实验室研究人员。发表了 10 多篇国内国际期刊/会议论文。研究方向为机器学习、大数据和网络安全。Email: yinjie@jspi.cn

张祎 于 2018 年在江苏警官学院网络安全与执法专业获得学士学位。现在南京大学图书情报(非全日制)专业攻读硕士学位, 现任江苏省公安厅网络安全保卫总队民警。研究领域为区块链数据分析、区块链监管科技。Email: mp21140047@ smail.nju.edu.cn

马卓 于 2021 年在东南大学网络空间安全专业获得工学博士学位, 现任江苏警官学院计算机信息与网络安全系讲师, 研究领域为时间序列分析、社交网络数据分析。Email: mazhuo@jspi.edu.cn