

基于GR算法的区块链智能合约漏洞检测

王德广, 单梦桃, 李凤岐, 佟宁
(大连交通大学软件学院, 辽宁 大连 116028)

摘要: 智能合约作为区块链技术的核心执行载体, 其安全性是确保链上数字资产有效管理的基石。针对当前智能合约漏洞检测存在覆盖漏洞类型有限、检测效率偏低的技术瓶颈, 提出重点解决多类型漏洞检测与特征提取效率优化两大核心问题。为此, 构建了包含3 000余条样本的5个专项数据集, 覆盖可重入漏洞、时间戳依赖漏洞、整数型溢出漏洞、交易顺序依赖和交易授权漏洞等主流漏洞类型, 为多种类检测奠定数据基础; 在算法设计层面, 提出改进的GR (gated recurrent unit- random forest) 算法模型, 其中改进型门控循环神经网络通过引入分离注意力机制强化关键漏洞特征的捕捉能力, 而随机森林算法则采用信息熵优化策略实现全局特征的完整性保留, 双通道处理架构既保证了关键漏洞特征的显著性提取, 又避免了深层特征在传递过程中的信息衰减, 旨在提升智能合约漏洞检测的种类数与效率。实验结果显示: GR算法模型能够识别出5种智能合约漏洞, 准确率高达98.88%。相较于先前算法模型, 检测效率提高3%以上, 漏洞类型平均增加3种, 验证了GR算法模型的可行性和优越性。

关键词: 区块链; 智能合约; 数据集预处理; 漏洞检测算法

中图分类号: TP393

文献标志码: A

DOI: 10.11959/j.issn.2096-109x.2025018

Blockchain smart contract vulnerability detection based on GR algorithm

WANG Deguang, SHAN Mengtao, LI Fengqi, TONG Ning
Software Technology Institute, Dalian Jiaotong University, Dalian 116028, China

Abstract: Smart contract, as the core execution mechanism in blockchain technology, have their security serving as

收稿日期: 2024-09-27; 修回日期: 2025-02-10

通信作者: 李凤岐, Fengqi-Li@outlook.com

基金项目: 辽宁省国际科技合作项目 (2022JH2/10700012); 辽宁省应用基础研究项目 (2023JH2/101300188, 2022JH2/101300269); 云南省服务计算重点实验室基金 (YNSC23118); 辽宁省教育厅基础研究项目 (JYTMS20230011)

Foundation Items: Liaoning Province International Science and Technology Cooperation Project (2022JH2/10700012), Liaoning Province Applied Basic Research Project (2023JH2/101300188, 2022JH2/101300269), Yunnan Provincial Key Laboratory Fund for Service Computing (YNSC23118), Basic Research Project of Liaoning Provincial Department of Education (JYTMS20230011)

引用格式: 王德广, 单梦桃, 李凤岐, 等. 基于GR算法的区块链智能合约漏洞检测[J]. 网络与信息安全学报, 2025, 11(2): 76-86.

Citation Format: WANG D G, SHAN M T, LI F Q, et al. Blockchain smart contract vulnerability detection based on GR algorithm[J]. Chinese Journal of Network and Information Security, 2025, 11(2): 76-86.

the cornerstone for the effective management of on-chain digital assets. To address the current technical bottlenecks in smart contract vulnerability detection, such as limited coverage of vulnerability categories and detection efficiency, the study was conducted. Two critical challenges were focused on: multi-category vulnerability detection and optimization of feature extraction efficiency. Methodologically, the following approaches were adopted: Five specialized datasets containing more than 3,000 annotated samples were constructed, covering mainstream vulnerability categories including reentrant vulnerabilities, timestamp dependency vulnerabilities, integer overflow vulnerabilities, transaction order dependence and transaction authorization vulnerabilities. Thus, a robust data foundation for multi-category detection was established. At the algorithmic design level, an improved GR (gated recurrent unit-random forest) algorithm model was proposed. In this model, the improved gated recurrent neural network (GRU) incorporated a decoupled attention mechanism to strengthen the ability to capture critical vulnerability features. Meanwhile, the random forest algorithm employed an information entropy optimization strategy to preserve the integrity of global features. The dual-channel processing architecture not only ensured the significant extraction of key vulnerability features, but also prevented information attenuation during deep feature transmission. Ultimately, the aim of improving both the variety and efficiency of smart contract vulnerability detection was achieved. Experimental results showed that the GR algorithm model could successfully detect five categories of smart contract vulnerabilities with an accuracy rate of 98.88%. Compared to previous algorithm models, it achieved over 3% improvement in detection efficiency and increased three detectable vulnerability categories. Thus, the feasibility and superiority of the GR algorithm model were validated.

Keywords: blockchain, smart contract, dataset preprocessing, vulnerability detection algorithm

0 引言

智能合约^[1]由 Szabo 提出,是在无可信第三方情况下,基于区块链技术的可自动执行条款的程序实现,具有传送、互动、存储数据等功能。以太坊^[2]是第一个支持智能合约的区块链平台,智能合约管理存储在区块链平台上的数字货币^[3],为方便智能合约的编译运行,以太坊为智能合约的执行构建了以太坊虚拟机(ethereum virtual machine, EVM)。Solidity^[4]语言是智能合约最常用的编写语言,以字节码的形式在以太坊上运行。随着智能合约的广泛应用,安全隐患也悄然而至。在 The DAO 事件^[5]中,攻击者多次调用转账函数,造成的损失高达 6 000 万美元。因此,对智能合约进行漏洞检测十分有必要。

针对智能合约的漏洞检测, Qian 等^[6]从智能合约为何受漏洞影响、安全漏洞类型和典型案例等方面进行了系统介绍。当前有基于符号执行、模糊测试、污点分析、形式化验证、机器学习 5 种智能合约漏洞检测技术。Securify^[7]是一个静态合约分析工具,通过符号执行构建合约关系图,从智能合约源码中提取信息来检测合约是否规范、是否有安全漏洞,但需要手动分类比较耗

时。Harvey 是一种灰盒模糊测试技术,由 Wüstholtz 等^[8]针对智能合约漏洞提出,通过预测出满足覆盖率高的路径来检测漏洞,但覆盖不全。Ethainter^[9]是一种基于污点分析的处理技术,对智能合约中的数据流进行分析,存在污点处理不全面的问题。VaaS^[10]是一个基于形式化验证的安全检测平台,其可以自动检测合约中的常规安全问题,并给出风险的代码位置和修改意见,但错误率较高。

近年来,机器学习在智能合约漏洞检测领域崭露头角,其良好的扩展性给漏洞检测提供了更多的可能性。文献[11]提出了 BNMC 模型,首次尝试构建贝叶斯网络模型,检测每个漏洞的严重程度,并提出漏洞的原因,其整体速度快,但可检测的漏洞类型少且准确率在 80% 左右。文献[12]提出了 BiLSTM-ATT 模型,通过获得的操作码词向量进行序列特征的提取,并使用注意力机制赋权,进行合约漏洞检测,整体效果有所提升,但存在检测准确率低和检测漏洞类型少的问题。文献[13]提出了 BiGRU-ATT 模型,主要针对无人机集群建设安全系数研究,提高了智能合约漏洞检测的准确率和精确率,但检测的漏洞类型少。

针对以上问题, 本文提出一种基于改进的门控循环神经网络与随机森林相结合的智能合约漏洞检测模型, 该模型通过特征融合, 获取更全面的特征, 提高了漏洞检测的准确率和种类。本文模型致力于检测出智能合约未被部署阶段的漏洞, 因此, 采用智能合约的源码进行检测, 提高对智能合约理解性的同时易于开发者修复检测出的漏洞。

本文模型把智能合约源码通过 Word2vec 技术转化为词向量, 再对词向量进行分析, 最后使用改进的 GRU (gated recurrent unit) 网络与 RF (random forest) 相融合的算法模型, 即 GR (gated recurrent unit-random forest) 算法模型对数据进行训练与测试。由于缺少直接送入机器学习算法的智能合约漏洞数据集, 本文构建了 SC-5 数据集, 可对可重入、时间戳依赖、整数型溢出、交易顺序依赖和交易授权等 5 种智能合约漏洞进行检测。实验结果表明, GR 算法模型在 5 种漏洞类型上检测准确率分别达到 98.88%、95.88%、96.36%、96.98%、97.60%。

综上所述, 本文所做贡献如下。

1) 针对 5 种智能合约漏洞分别构建了 5 个具有 3 077 条数据的漏洞数据集, 将其命名为 SC-5 数据集, 以便后续算法使用。

2) 改进了 GRU 网络, 在 GRU 网络中融合了分离注意力机制, 对输入数据中的重要特征进行加权, 突出关键信息, 提高了智能合约检测的准确率和效率。

3) 提出了 GR 算法模型, 将改进的 GRU 网络和随机森林两种算法相结合, 以更全面地获取数据特征, 提高了合约的漏洞检测类型。将本文提出的 GR 算法模型在 SC-5 数据集上进行验证, 结

果显示准确率可达 98.88%, 验证了本文所提算法模型的可行性和高效性。

1 相关理论

1.1 智能合约安全漏洞

智能合约截至 2024 年仅在以太坊上管理资金已超过千亿美元, 因其一旦部署便不可篡改性, 在智能合约被部署之前进行漏洞检测显得十分重要。目前, 智能合约漏洞有 20 多种, 本文主要针对智能合约中的 5 种漏洞进行检测, 智能合约漏洞分类及相关信息如表 1 所示。

1.2 Word2vec 技术

Word2vec^[14]是一种用于生成词向量的技术, 由 Google 团队在 2013 年提出, 可以在大量未标注的文本数据上快速高效地训练词向量模型。Word2vec 是由输入层、隐藏层和输出层组成的轻量级网络, 根据输入输出的形式不同, 可分为 CBOW 和 Skip-gram 两种模型。其中, CBOW 模型是从上下文词中提取信息, 来预测当前词; Skip-gram 模型是由当前词预测上下文。两个模型都使用一个浅层神经网络, 输入层用 one-hot 向量来表示词, 经过隐藏层转换为低维向量, 输出层用于预测目标值。

1.3 随机森林

随机森林^[15]是一种采用 Bagging 思想的集成算法, 以决策树为基本单元, 通过集成大量的决策树构成, 将数据分类, 再以投票的形式得出平均值作为最终的结果。即每次从训练集中取 a 个训练样本, 以有放回的形式反复进行, 得到一个全新的训练集; 再利用新的训练集, 经过训练形成 b 个子模型。针对分类的问题, 最终的类别是得到票数最多的子模型的类别。随机森林构造过程如图 1 所示。

表 1 智能合约漏洞分类及相关信息

Table 1 Classification and related information of vulnerabilities in smart contracts

漏洞类型	漏洞原因	相关攻击	安全问题
可重入漏洞	Fallback 函数中存在对外部合约函数的递归调用	The DAO 攻击	无法存储和保护合约代币或数据
时间戳依赖漏洞	区块时间戳被赋值给可预测的变量	—	无法使用安全的随机数
整数型溢出漏洞	数值超出或低于定义的整数类型范围	美链 BEC	整数范围错误
交易顺序依赖	交易顺序不一致	—	竞争条件或非法预先交易
交易授权漏洞	tx.origin 全局变量用于智能合约身份验证	—	不适合的合约或函数访问

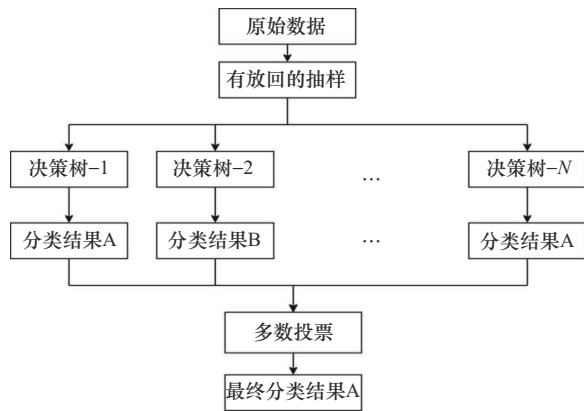


图1 随机森林构造过程

Figure 1 Random forest construction process

1.4 GRU 网络结构

门控循环神经网络^[16]与RNN^[17]、LSTM^[18]类似，只有更新门和重置门两个结构。GRU更新门决定过去信息和当前输入的权重，控制过去信息的保留和更新；重置门决定过去记忆对当前输入的影响，控制过去记忆的遗忘和重置。GRU模型结构如图2所示。

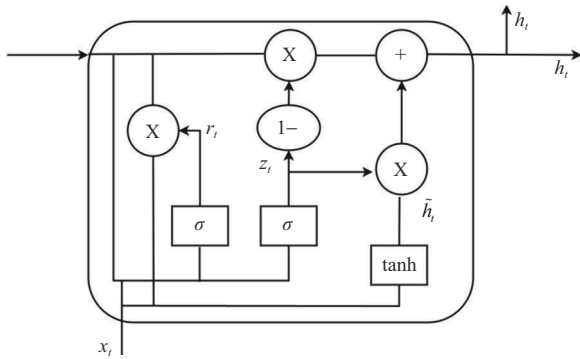


图2 GRU 模型结构

Figure 2 Structure of the GRU model

利用前一时刻隐藏层输出状态 h_{t-1} 和现有节点输入 x_t 建立更新门和重置门，其值越大，表示忽略的信息越少。计算式如下。

$$z_t = \sigma(W_z[h_{t-1}, x_t]) \quad (1)$$

$$r_t = \sigma(W_r[h_{t-1}, x_t]) \quad (2)$$

$$\tilde{h}_t = \tanh(W_h[r_t h_{t-1}, x_t]) \quad (3)$$

$$h_t = (1 - z_t)h_{t-1} + z_t \tilde{h}_t \quad (4)$$

式中： z_t 表示更新门， r_t 表示重置门。 σ 指的sig-

moid函数和tanh函数都是激活函数， x_t 指当前时刻的输入。 h_t 指当前时刻隐藏层的输出， h_{t-1} 指前一时刻隐藏层的输出， $\tilde{h}_t$ 指候选隐藏层状态。

2 GR 算法模型

智能合约漏洞检测框架如图3所示。智能合约漏洞检测模型由智能合约数据集处理层、GR算法模型层、漏洞检测层和模型评估层4个部分组成。

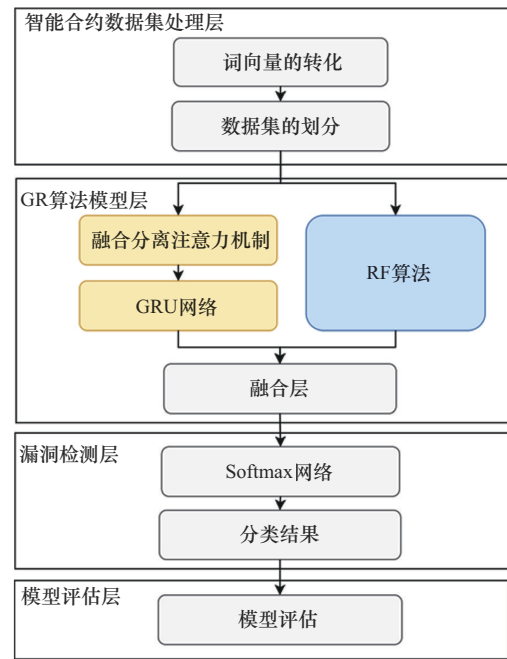


图3 智能合约漏洞检测框架

Figure 3 Smart contract vulnerability detection framework

2.1 GR 算法模型层

GR 算法模型层由融合分离注意力机制的GRU网络、RF算法及融合层组成，算法提取的特征在融合层进行相加，得到最终的漏洞特征，再将其送到漏洞检测层。

2.1.1 融合分离注意力机制的GRU网络

在整个智能合约源码中，漏洞语句只占关键的小部分，但把整个源码的特征信息输入GRU网络时，是对整个智能合约进行特征提取，没有对关键的漏洞语句进行侧重。为了更加突出漏洞信息的特征，本文在GRU网络中引入了分离注意力机制的算法结构，构建多通道结构，并为每个通道配置不一样的权重，以提高在处理序列数据时对关键信息的关注度，改进

后的 GRU 模型如图 4 所示。经过加权的数据信息重塑为三维张量送入 GRU 网络中, 分别经过两层 GRU 网络, 采用 ReLU 激活函数进行特征提取。

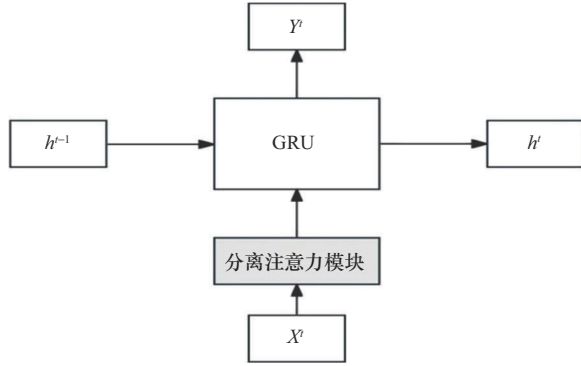


图 4 改进的 GRU 模型

Figure 4 Improved GRU model

分离注意力模块对输入的 50 维智能合约词向量进行权重分配, 把输入的各个序列分别作为一个单元进行计算, 将获得的系数与输入特征的相应元素相乘, 形成新的特征。这些新特征就是加权特征, 可以使模型更关注有用的信息, 以增强模型对漏洞特征的注意力分配。也就是说, 分离注意力模块起到突出漏洞特征信息的作用。分离注意力模块的算法结构如图 5 所示。

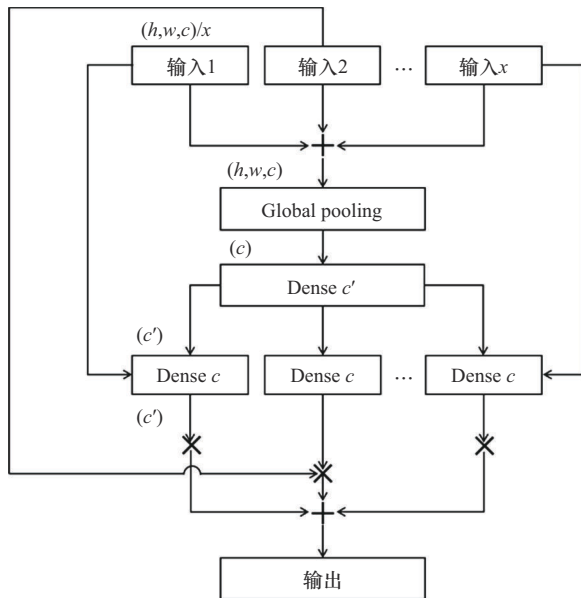


图 5 分离注意力算法结构

Figure 5 Algorithm structure of separation attention module

随后, 将加权的特征送入 GRU 网络中, 利用 GRU 的更新门和重置门对输入的词向量进行特征提取, 为上下文向量提供更丰富有用的信息, 以此提高 GRU 网络处理智能合约词向量序列数据的性能。

2.1.2 RF 算法

GR 算法模型中的 RF 算法是直接将转化来的 50 维词向量作为输入序列进行特征学习, 创建 RandomForestClassifier 模型, 把对应的 one-hot 标签作为目标变量进行训练, 从而保留了原始数据的全部信息, 起到平衡漏洞数据集因漏洞数据占比少导致误差的作用。划分属性时本文选择了基尼指数, 其值的大小表示数据集的纯度, 值越小, 纯度越高。基尼指数的定义如式(5)所示。

$$\text{Gini}(D) = 1 - \sum_{k=1}^K \left(\frac{|C_k|}{|D|} \right)^2 \quad (5)$$

在给定的样本集合 D 中, K 指类的个数, C_k 则表示 D 中属于第 k 类的样本子集。

2.1.3 融合层

为更好地检测智能合约漏洞, GR 算法模型采用改进的 GRU 网络与 RF 算法并行计算特征, 随后进行特征融合的思路。一是融合分离注意力机制的 GRU 网络, 对智能合约中的漏洞语句进行加权, 使得检测算法有所侧重; 二是采用 RF 算法对原始的 50 维词向量进行特征学习, 平衡了漏洞数据集导致的误差, 使得数据信息更加全面、准确。将两个算法模型提取的特征送入融合层, 采用直接相加的方式进行特征融合, 两个算法相辅相成以充分挖掘特征之间的相关性, 增强模型对智能合约漏洞的判别力。最后, 使用 Dropout 层缓解过拟合, Softmax 层将样本数据分为有漏洞和无漏洞两类。

分离注意力机制模块突出重要信息, 结合 GRU 网络有效抑制梯度消失的功能, 更适用于捕捉长序列语义关联的场景, 还通过 RF 算法进行特征选取, 加强了整体特征选择的表达。提取得到的两种特征在融合层相加, 使得 GR 算法模型更全面地获取特征信息, GR 算法模型的优点如表 2 所示。

表2 GR算法模型的优点

Table 2 The advantages of the GR algorithm model

模型	处理序列数据能力	处理不平衡数据	突出关键信息
BNMC ^[11]	×	×	×
BiLSTM-ATT ^[12]	√	×	√
BiGRU-ATT ^[13]	√	×	√
GR算法	√	√	√

2.2 漏洞检测层

漏洞检测层把经过融合层相加后的特征，即改进的GRU网络捕捉上下文信息学习到的漏洞特征、逻辑与RF算法得到的漏洞特征相加，送入Softmax函数，利用概率分布得出漏洞检测结果。Softmax函数的基本概念如式(6)所示，其中 Z_i 指第 i 个节点的输出值， C 指分类的类别个数，使用Softmax函数将输出值转换为0到1之间的概率分布。本文将阈值设置为0.5，当值大于等于0.5时，分类结果将其判断为有漏洞；值小于0.5时，判断为无漏洞。

$$\text{Soft max}(z_i) = \frac{e^{z_i}}{\sum_{c=1}^C e^{z_c}} \quad (6)$$

2.3 分类算法评价指标

选择准确率(accuracy)、精确率(precision)、召回率(recall)和F1分数(F1_score)4个指标来评估模型的测试结果。指标是根据表3的混淆矩阵(confusion matrix)计算得来的。其中，准确率是模型预测与实际情况相符的数量占总量的比例，可以评估模型对整个数据集的分类情况；精确率是实际有漏洞的样本占预测有漏洞样本的比例，可以评估模型对漏洞的误判情况；召回率是预测有漏洞的样本占实际有漏洞样本的比例，可以评估模型对真实漏洞识别的能力；F1分数同时考虑了准确率和召回率，可以很好地反映模型的检测效果。

表3 混淆矩阵

Table 3 Confusion matrix

检测值	真实值	
	有漏洞	无漏洞
有漏洞	真正例 (TP)	假正例 (FP)
无漏洞	假反例 (FN)	真反例 (TN)

如表3所示，TP指真实漏洞被预测为有漏洞的样本数量，FN指真实漏洞被预测为无漏洞的样本数量，FP指真实无漏洞样本预测为有漏洞样本的数量，TN指真实无漏洞样本预测为无漏洞的数量。

通过计算得到相对应的准确率、召回率、精确率和F1分数，计算式分别如下。

$$\text{accuracy} = \frac{\text{TP} + \text{TN}}{\text{TP} + \text{TN} + \text{FP} + \text{FN}} \quad (7)$$

$$\text{precision} = \frac{\text{TP}}{\text{TP} + \text{FP}} \quad (8)$$

$$\text{recall} = \frac{\text{TP}}{\text{TP} + \text{FN}} \quad (9)$$

$$F1_score = \frac{2 \times \text{precision} \times \text{recall}}{\text{precision} + \text{recall}} \quad (10)$$

3 构建SC-5数据集

智能合约的漏洞检测大体可分为传统的智能合约漏洞检测和基于深度学习的漏洞检测。传统的智能合约漏洞检测方法(如Yamaguchi等^[19])结合抽象语法树、程序依赖图或通过反编译合约字节码构建控制流图，再用约束求解算法进行约束求解等，运用传统动态或静态的方法进行检测。基于深度学习的智能合约漏洞检测方法能够取得突破性的进展依赖于统一且全面的智能合约漏洞数据集^[6]。目前，智能合约漏洞领域没有统一可直接送入算法的数据集。为顺利将智能合约源码送入网络，本文针对可重入、时间戳依赖、整数型溢出、交易顺序依赖和交易授权5种智能合约漏洞，分别构建了5个漏洞数据集，统称为SC-5数据集，该数据集可以送入模型进行漏洞检测。

3.1 词向量

本文使用Etherscan浏览器获得智能合约源码，并采用Word2vec技术中的CBOW模型对源码进行处理。CBOW模型有输入层、隐藏层和输出层3个层级。具体的模型架构如图6所示，设定window(窗口值)=4， $W(t)$ 代表中心词，则 $W(t-4)$ 、 $W(t-3)$ 、 $W(t-2)$ 、 $W(t-1)$ 、 $W(t+1)$ 、 $W(t+2)$ 、 $W(t+3)$ 、 $W(t+4)$ 为 $W(t)$ 的上下文。

在输入层，将所有上下文单词的one-hot编

码作为输入，并假设单词向量空间的维度为 V ，所以每一个单词都有一个编号 $i \in \{1, \dots, |V|\}$ ，并表示成一个维度为 $|V|$ 的词向量空间，其中第 i 个元素值非 0，余下的是值为 0 的元素；在隐藏层中，因为输入层到隐藏层之间有一个权重矩阵 W ，隐藏层得到的值是由输入值与权重矩阵的乘积。如果隐藏层最终得到一个 N 维度大小的词向量 N ，输入层与隐藏层间的权重矩阵为 $W \in \mathbb{R}^{V \times N}$ ，则 $V \times N$ 就是 W 的大小；接着是输出层，由于在隐藏层和输出层间存在一个权重矩阵 W' ，则输出层向量 y 的每一个值，是隐藏层向量与权重 W' 对应列向量的点积运算结果。最后通过对 Softmax 函数做归一化处理，把计算结果中的每一个元素都转化成分布在 0 至 1 的概率数值，其中值最大的是由预测得到的结果。

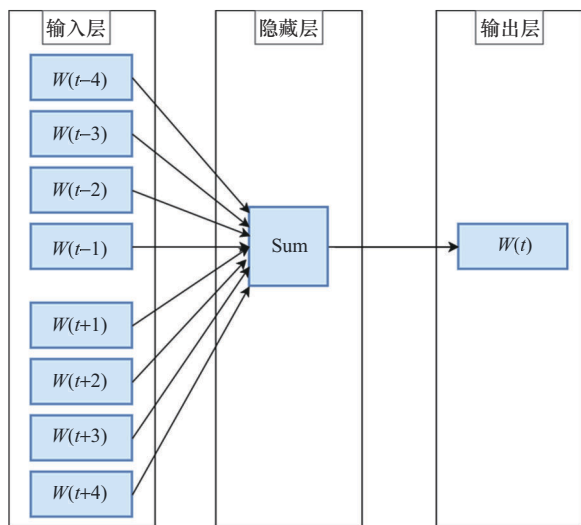


图6 CBOW模型架构

Figure 6 CBOW model architecture

3.2 词向量的转化过程

运用 CBOW 模型把智能合约源码转化为词向量，再手动区分转化出来的词向量是否包含漏洞，打上 one-hot 标签，最后经过去重等操作转换成可送入算法的智能合约漏洞的 SC-5 数据集。

在转化为词向量的过程中，首先尝试 100 维的词向量，因转化维度较大，转化出来的词向量包含关键信息量占比较少，导致在训练过程中出现过拟合现象。接着逐次尝试转化为 75 维、50 维、25 维的情况。在 75 维时过拟合现象有所好转，但训练结果不理想；50 维时效果最佳，而且出现

词向量条重复的现象，可作为转化出源代码所有信息的依据，可将重复词向量条去除得到完整的源码信息；25 维时出现更多的重复语句，但由于词向量维度较小，每条词向量的信息完整度较低，训练得到的效果不理想。通过实验验证，本文选择将源码转化为 50 维的词向量效果最佳，词向量维度参数信息如表 4 所示。

表4 词向量维度参数信息

Table 4 Information of word vector dimension parameter

转化维度数	原因	效果
100 维	原始信息丢失	过拟合
75 维	减少信息丢失	好转
50 维	原始信息完整	最佳
25 维	每条信息不完整	效果差

3.3 SC-5数据集

本文针对 5 种漏洞类型，构建 SC-5 数据集，如表 5 所示。

表5 SC-5数据集

Table 5 SC-5 dataset

数据集漏洞类型	总数据条/条	包含漏洞数/条	未有漏洞数/条
可重入	3 077	453	2 624
时间戳依赖	3 077	441	2 636
整数型溢出	3 077	409	2 668
交易顺序依赖	3 077	413	2 664
交易授权	3 077	413	2 664

对数据进行 one-hot 编码，用 0、1 标签来标记每一条词向量是否具有漏洞，构建出 SC-5 数据集。SC-5 数据集中包含 5 种漏洞类型，针对每一种智能合约漏洞有 3 077 条数据，其中漏洞数据条数是 400 多条，无漏洞数据是 2 600 多条，整体有无漏洞数据比约为 1:6。

4 实验结果及分析

4.1 实验环境

本文实验使用深度学习的 Keras 框架，运行环境为 Windows 11，硬件平台为 AMD Ryzen 7 5800H with Radeon Graphics 3.20 GHz，16 GB RAM，NVIDIA RTX3060，软件环境的配置为 keras2.6、tensorflow2.6、cuDNN8.1、CUDA11.2，

python3.9语言，编译环境为jupyter notebook。

4.2 模型参数设置

实验参数的选择严重影响预测结果，本文从词向量维度、学习率、批量大小、训练轮次、隐藏层特征数量和减小过拟合等参数进行寻优，实验参数最终设定值如表6所示。

其中，词向量维度在构建数据集时确定转换为50维。初始学习率分别尝试设置为0.01、0.001、0.000 1。值为0.01时训练效果不理想，0.001时训练效果较好，0.000 1效果变化不明显，综合考虑初始学习率设置为0.001。批量大小设置过大或过小都会对模型性能、训练时长有影响，本文设置批量值为16、32、64进行实验，最终确定批量大小值为32。模型的拟合能力受训练轮次数影响，轮次过多时易出现过拟合现象，设置为50、100、150、200、300轮进行实验，结果显示在100到200轮次时表现最优，300轮次时结果有所下降，综合考虑，最终设置为100轮次。隐藏层特征数量的设置，影响函数的拟合效果，原则上数量越多拟合效果好，但实际上过多的特征数会带来过拟合、训练难度增加等问题，本文选取了16、32、64进行实验，其中32时效果最优。丢弃法是深度学习常用方法，原理是在神经网络训练过程中，随机删掉所设置比例的神经元个数以防止过拟合，尝试随机删除0.1、0.2、0.3，综合实验效果考虑，最终设置丢弃值为0.2。

4.3 本文模型训练过程

在SC-5数据集中，80%的数据被随机划分为训练集，其余为测试集，测试集中的20%被划分为验证集。图7为本文GR算法模型针对可重入、时间戳依赖、整数型溢出、交易顺序依赖和

交易授权等5种漏洞的检测结果，检测的准确率分别为98.88%、95.88%、96.36%、96.98%和97.60%，整体表现良好。其中，可重入漏洞准确率最高，时间戳漏洞准确率最低，主要原因是可重入发生在合约函数调用外部合约的过程中，漏洞代码较明显，而时间戳依赖涉及复杂的逻辑条件，不易提取出明确的漏洞模式，因而检测准确率较低。提高逻辑复杂场景下的识别度是漏洞检测准确率的难点，也是今后工作的重点。图7中红线和绿线、灰线和蓝线整体走向趋近，说明训练过程中没出现过拟合现象。

4.4 对比实验

为检验本文模型的性能，将本文工作分别与表现较好的RF、CNN、GRU、BNMC^[11]、BLSTM-ATT^[12]、BiGRU-ATT^[13]等6种模型进行对比，结果见表7。

由表7可以看出，RF、CNN和GRU网络均支持检测5类漏洞，其中CNN算法表现较差；RF算法因其可以处理高维度数据和平衡不平衡数据集的误差，表现良好；GRU网络因其可以利用循环结构保存上下文信息，并通过遗忘门和更新门保留更有用的信息，实验结果良好。文献[11]不支持时间戳依赖、整数型溢出和交易顺序依赖等漏洞的检测，文献[12]不支持检测交易顺序依赖和交易授权漏洞，文献[13]不支持检测整数型溢出、交易顺序依赖和交易授权漏洞，本文支持检测5种常见的漏洞。由此可知，本文模型在检测漏洞种类上有所提升。

在检测漏洞的准确率上，本文模型表现良好。对于可重入漏洞，本文模型较对比模型中最高的准确率、召回率、精确率、F1分数分别提高了2.39%、11.05%、1.33%、33.40%。对于时间戳依

表6 实验参数
Table 6 Experimental parameters

参数	参数含义	设定值范围	最终设定值
Embed_size	转化的词向量的维度	25、50、75、100	50
Lr	设置的初始学习率	0.01、0.001、0.000 1	0.001
Batch_size	设置每个批次的训练样本数	16、32、64	32
Epoch	训练轮次	50、100、150、200、300	100
Num_hiddens	设置的隐藏层特征数量	16、32、64	32
Dropout	丢弃神经元	0.1、0.2、0.3	0.2

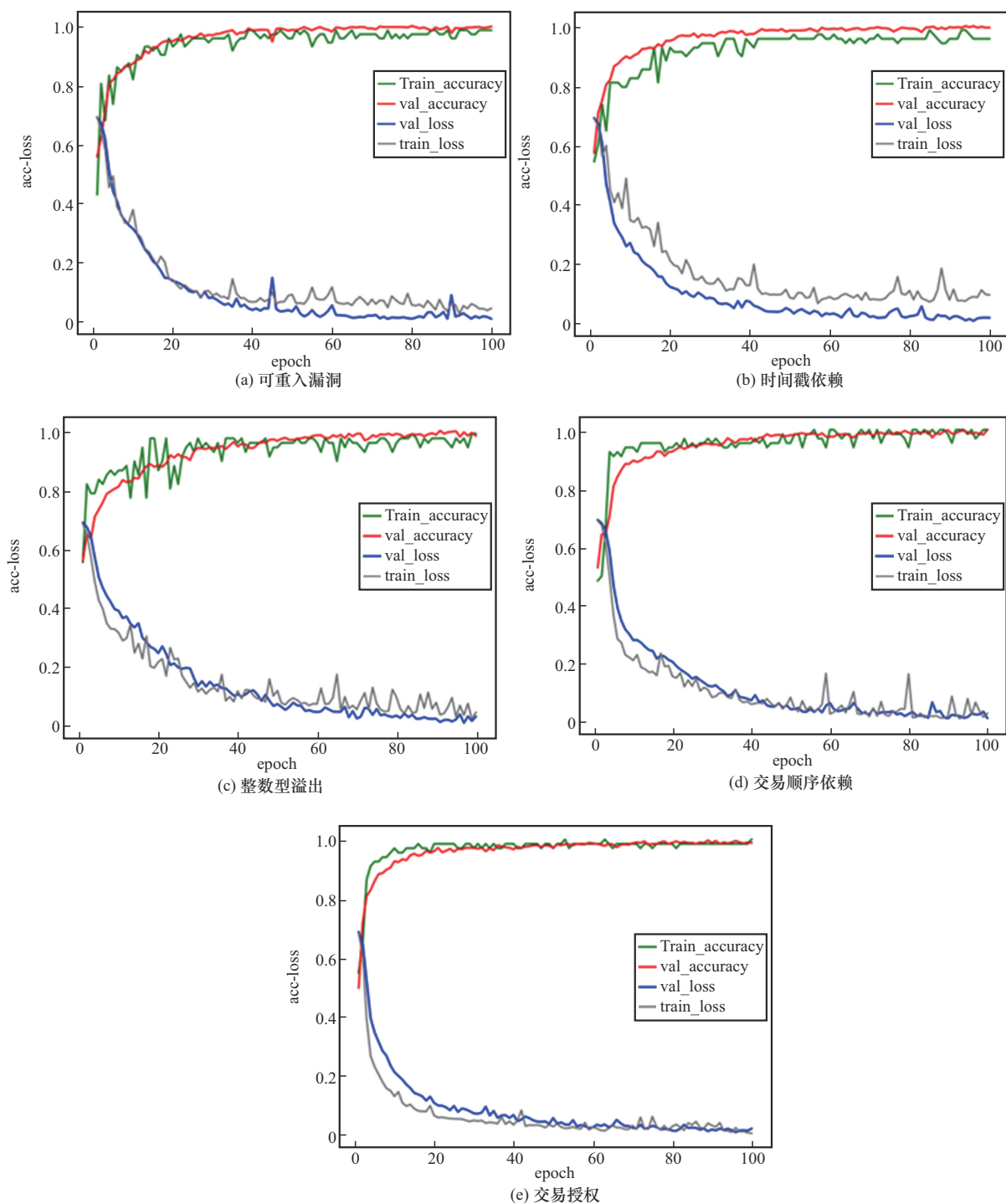


图7 GR算法模型对5种漏洞的检测结果

Figure 7 The detection results of GR model for 5 types of vulnerabilities

赖, 本文模型的准确率和召回率虽没有文献[13]高, 但精确率和 $F1$ 分数较最高的模型, 分别提高了5.45%、17.58%。对于整数型溢出, 本文模型较对比中最高模型, 召回率提高了4.79%、精确率提高了1.74%、 $F1$ 分数提高了3.17%, 准

确率在对比文献中没有体现, 本文模型为96.36%。对于交易顺序依赖, 对比的文献中没有检测此漏洞, 本文模型的准确率、召回率、精确率和 $F1$ 分数分别为96.98%、96.97%、97.00%、96.98%。对于交易授权漏洞, 本文模型的准确

表7 各模型在5种漏洞类型上的对比结果

Table 7 Comparison results of model on 5 types of vulnerabilities

检测模型	指标	可重入	时间戳依赖	整数型溢出	交易顺序依赖	交易授权
RF	准确率	92.14%	90.26%	87.36%	88.06%	90.28%
	精确率	92.35%	91.03%	86.31%	88.38%	89.36%
	F1 分数	91.79%	90.87%	86.46%	89.28%	89.89%
CNN	准确率	77.09%	78.21%	74.49%	64.24%	62.78%
	精确率	77.71%	70.94%	73.23%	64.48%	63.12%
	F1 分数	78.85%	75.41%	71.65%	63.97%	62.88%
GRU	准确率	93.34%	89.94%	90.88%	89.12%	92.31%
	精确率	92.17%	90.62%	88.26%	89.35%	91.29%
	F1 分数	92.19%	90.00%	89.15%	89.26%	92.87%
文献[11]	准确率	88.61%	—	—	—	78.61%
	召回率	87.78%	—	—	—	81.67%
	精确率	89.27%	—	—	—	76.96%
文献[12]	召回率	76.39%	73.97%	91.50%	—	—
	精确率	93.18%	83.08%	94.92%	—	—
	F1 分数	65.48%	78.26%	93.18%	—	—
文献[13]	准确率	96.49%	96.18%	—	—	—
	召回率	87.80%	97.90%	—	—	—
	精确率	97.60%	90.20%	—	—	—
GR	准确率	98.88%	95.88%	96.36%	96.98%	97.60%
	召回率	98.85%	96.19%	96.29%	96.97%	97.61%
	精确率	98.93%	95.65%	96.66%	97.00%	97.62%
	F1 分数	98.88%	95.84%	96.35%	96.98%	97.60%

率、召回率和精确率较对比中最高的模型分别提高了18.99%、15.94%、20.66%，F1分数在对比文献中没有体现，本文模型为97.60%。

5 结束语

由于智能合约的安全十分重要，且部署后不可更改的特性，本文针对5种类型的智能合约漏洞，在合约源码被部署到以太坊之前对其进行漏洞检测。本文提出了一种基于改进的GRU网络融合RF算法的模型，实现多种类型的智能合约漏洞检测。从对比实验可以看出，本文算法模型在检测漏洞的准确率、检测的漏洞类型种类上均有所提高，尤其是准确率可达98.88%，验证了本文模型在性能方面的优越性。在今后工作中，将会继续扩建智能合约漏洞数据集的漏洞数据量和漏洞种类，或使用小样本算法对智能合约进行

检测，进一步提高算法的准确性和检测合约漏洞的种类数。

参考文献：

- [1] SZABO N. Smart contracts: building blocks for digital markets[J]. EXTROPY: The Journal of Transhumanist Thought, 1996, 18(2): 28.
- [2] CHEN J, XIA X, LO D, et al. Defectchecker: Automated smart contract defect detection by analyzing evm bytecode[J]. IEEE Transactions on Software Engineering, 2021, 48(7): 2189-2207.
- [3] 张娴, 朱建明, 隋智源等. 数字货币交易匿名性与监管的博弈分析[J]. 网络与信息安全学报, 2022, 8(5): 150-157.
ZHANG X, ZHU J M, SUI Z Y, et al. Game analysis of anonymity and regulation in digital currency transactions[J]. Journal of Network and Information Security, 2022, 8(5): 150-157.
- [4] NGUYEN T D, PHAM L H, SUN J, et al. Sfuzz: An efficient adaptive fuzzer for solidity smart contracts[C]//Proceedings of the ACM/IEEE 42nd International Conference on Software Engineering. 2020:

- 778-788.
- [5] MEHAR M I, SHIER C L, GIAMBATTISTA A, et al. Understanding a revolutionary and flawed grand experiment in blockchain: The DAO attack[J]. Journal of Cases on Information Technology (JCIT), 2019, 21(1): 19-32.
- [6] 钱鹏, 刘振广, 何钦铭, 等. 智能合约安全漏洞检测技术研究综述[J]. 软件学报, 2022, 33(8): 3059-3085.
- QIAN P, LIU Z G, HE Q M, et al. Review of security vulnerability detection technologies for smart contracts[J]. Journal of Software, 2022, 33(8): 3059-3085.
- [7] TSANKOV P, DAN A, DRACHSLER-COHEN D, et al. Securify: Practical security analysis of smart contracts[C]//Proceedings of the 2018 ACM SIGSAC Conference on Computer and Communications Security. 2018: 67-82.
- [8] WÜSTHOLZ V, CHRISTAKIS M. Harvey: A greybox fuzzer for smart contracts[C]//Proceedings of the 28th ACM Joint Meeting on European Software Engineering Conference and Symposium on the Foundations of Software Engineering. 2020: 1398-1409.
- [9] BRENT L, GRECH N, LAGOUVARDOS S, et al. Ethainter: A smart contract security analyzer for composite vulnerabilities[C]//Proceedings of the 41st ACM SIGPLAN Conference on Programming Language Design and Implementation. 2020: 454-469.
- [10] 张文博, 陈思敏, 魏立斐, 等. 基于形式化方法的智能合约验证研究综述 [J]. 网络与信息安全学报, 2022, 8 (4): 12-28.
- ZHANG W B, CHEN S M, WEI L F, et al. State-of-the-art survey of smart contract verification based on formal methods[J]. Chinese Journal of Network and Information Security, 2022, 8(4): 12-28.
- [11] KODAVALI L, KUPPUSWAMY S. Bayesian network based vulnerability detection of blockchain smart contracts[J]. Journal of Intelligent & Fuzzy Systems, 2023, 44(2): 1907-1920.
- [12] 张光华, 刘永升, 王鹤, 等. 基于 BiLSTM 和注意力机制的智能合约漏洞检测方案[J]. 信息安全学报, 2022, 22 (9): 46-54.
- ZHANG G H, LIU Y S, WANG H, et al. A smart contract vulnerability detection scheme based on BiLSTM and attention mechanism[J]. Information Network Security, 2022, 22 (9): 46-54.
- [13] 杨忠举, 朱卫星, 何红悦, 等. 基于智能合约的无人机集群安全性研究[J]. 软件导刊, 2023, 22 (8): 164-171.
- YANG Z J, ZHU W X, HE H Y, et al. Research on the security of drone clusters based on smart contracts[J]. Software Guide, 2023, 22 (8): 164-171.
- [14] SUN W, MA F, TZACHOR A, et al. Bridging environmentally extended input-output models and complex network analysis: A bibliometric analysis of trends and opportunities[J]. Journal of Cleaner Production, 2025, 486 144427-144427.
- [15] SUN X R, CHAI J. Random forest feature selection for partial label learning [J]. Neurocomputing, 2023, 561.
- [16] ZHANG Y, ZHOU W, HUANG L, et al. Efficient microbubble Trajectory tracking in ultrasound localization microscopy using a gated recurrent unit-based multitasking temporal neural network[J]. IEEE Transactions on Ultrasonics, Ferroelectrics, and Frequency Control, 2024.
- [17] SHERSTINSKY A. Fundamentals of recurrent neural network (RNN) and long short-term memory (LSTM) network[J]. Physica D: Nonlinear Phenomena, 2020, 404: 132306.
- [18] GAO S, HUANG Y, ZHANG S, et al. Short-term runoff prediction with GRU and LSTM networks without requiring time step optimization during sample generation[J]. Journal of Hydrology, 2020, 589: 125188.
- [19] YAMAGUCHI F, GOLDE N, ARP D, et al. Modeling and discovering vulnerabilities with code property graphs[C]//2014 IEEE Symposium on Security and Privacy. 2014: 590-604.

[作者简介]



王德广 (1968-), 男, 辽宁大连人, 大连交通大学副教授, 主要研究方向为信息安全、机器学习。



单梦桃 (1997-), 女, 河南驻马店人, 大连交通大学硕士生, 主要研究方向为区块链、智能合约漏洞、机器学习。



李凤岐 (1974-), 男, 河北承德人, 大连交通大学教授、博士生导师, 主要研究方向为区块链技术、工业互联网、大数据与智能计算。



佟宁 (1981-), 女, 辽宁盘锦人, 大连交通大学副教授, 主要研究方向为无线网络安全、人工智能、区块链。