基于图神经网络的账户余额模型区块链地址分类方法

李致远[1,2,3]，徐丙磊[1]，周颖仪[1]

（1. 江苏大学计算机科学与通信工程学院，江苏 镇江 212013；
2. 江苏省工业网络安全技术重点实验室，江苏 镇江 212013；
3. 江苏省泛在数据智能感知与分析应用工程研究中心，江苏 镇江 212013）

摘 要：为了监管账户余额模型公链上的交易，有必要对该类区块链上的交易进行地址分类研究。基于此，提出了一种基于图神经网络的账户余额模型区块链地址分类方法（简称 AJKGS-ABCM）以实现区块链地址的分类，为区块链交易追踪提供有效的支持。该方法将区块链交易数据建模为图结构，以地址为节点，交易为边，提出 AJK-GraphSAGE 算法学习图的嵌入表示，模型的输入只需要节点及其采样的邻居节点集合。同时，模型引入注意力机制及跳跃知识结合策略，自适应地为不同层的表示分配权重，并在不同层间共享信息，提高了训练速度和泛化能力。最后进行了实验对比，结果表明该模型在准确度、召回率和 F1 分数上性能优于其他方法。

关键词：账户余额模型区块链；地址分类；图神经网络；注意力机制；跳跃知识

中图分类号：TP301

文献标志码：A

DOI: 10.11959/j.issn.1000−436x.2023173

Graph neural network-based address classification method for account balance model blockchain

LI Zhiyuan[1,2,3], XU Binglei[1], ZHOU Yingyi[1]

1. Jiangsu University, School of Computer Science and Communication Engineering, Zhenjiang 212013, China
2. Jiangsu Provincial Key Laboratory of Industrial Network Security Technology, Zhenjiang 212013, China
3. Jiangsu Province Ubiquitous Data Intelligent Perception and Analysis Application Engineering Research Center, Zhenjiang 212013, China

Abstract: To regulate the transactional activities on the public blockchain involving account balance models, it is necessary to conduct research on address classification for transactions on such blockchains. A blockchain address classification method, named AJKGS-ABCM (attention jumping knowledge graph SAGE account-based blockchain classification model), was proposed to categorize blockchain addresses, providing effective support for blockchain transaction tracking. Blockchain transaction data was represented as a graph structure, with addressed as nodes and transactions as edges. The AJK-GraphSAGE algorithm was introduced to learn embedded representations of the graph, where the model's input required only nodes and their sampled neighboring node sets. Simultaneously, attention mechanisms and skip-connection knowledge integration strategies were incorporated into the model, allowing for adaptive weight allocation across different layers and information sharing between various levels, thereby enhancing training speed and generalization capabilities. Finally, experimental comparisons are conducted, demonstrating superior performance in terms of accuracy, recall, and F1 score compared to other methods.

Keywords: account balance model blockchain, address classification, graph neural network, attention mechanism, jumping knowledge

收稿日期：2023−05−18；**修回日期：**2023−08−31

基金项目：国家重点研发计划基金资助项目（No.2020YFB1005503）；江苏省自然科学基金资助项目（No.BK20201415）

Foundation Items: The National Key Research and Development Program of China (No.2020YFB1005503), The Natural Science Foundation of Jiangsu Province (No.BK20201415)

0 引言

区块链[1]是一种融合对等网络、密码学和共识协议的分布式数据库技术。近年来，我国将区块链技术作为国家重点战略发展。区块链在金融、物流、管理、医疗等方面的应用蓬勃发展。

区块链的匿名性意味着参与者都是以匿名地址交互的，交易不需要身份验证，没有一个可信第三方为用户提供注册和认证，只需要确认对方的链上地址，这也被称为区块链隐私保护。然而，区块链匿名性是一把双刃剑，极易被网络犯罪所利用。对区块链交易实现匿名可追踪监管是重要且必需的。它首要解决的是交易地址的去匿名化问题，这可以识别异常交易行为，为交易跟踪提供更多参考，提高整个系统的透明度和可审计性。

当前，公链上的交易模型主要包括未消费交易输出（UTXO, unspent transaction output）模型和账户余额模型。UTXO 模型的代表应用是比特币及其分叉的数字货币；账户余额模型的代表应用是以太坊及其变种的数字货币。目前，针对 UTXO 模型区块链系统的地址分类研究已经相对成熟[2-9]。但是这类研究无法直接应用在账户余额模型区块链的去匿名化上。此外，支持 ERC20 协议的以太坊交易规模和数据量也远超比特币，且获取和分析以太坊交易网络的全节点数据也更困难。与比特币丰富易获取的数据集不同，目前只有少量公开的以太坊标签数据集，这也为实现以太坊地址的分类增加了难度。

现有的地址分类算法主要采用人工特征工程、图建模和图数据挖掘等理论方法，这些方法虽然有效，但也存在不足和挑战。一方面，由于区块链平台间的技术差异，基于人工特征的算法在不同平台之间的泛用性较弱；另一方面，区块链交易图中账户地址和交易规模巨大，导致基于随机游走的图嵌入算法或图神经网络（GNN, graph neural network）的分类模型在内存和时间上消耗较大。与此同时，区块链上交易量的不断增加，导致交互图在节点和边上频繁更新，这不利于基于图卷积网络等全图学习的算法。

本文主要研究工作如下。

1) 提出一种基于图神经网络的多分类模型 AJKGS-ABCM，用于对账户余额模型区块链地址进行分类，本文以账户余额模型中最典型的以太坊为目标，进行地址分类任务。

2) 基于以太坊数据的大规模性以及复杂图结构问题，所提模型结合 GraphSAGE、注意力机制和跳跃知识结合策略，能够自适应地聚合邻居节点的信息，使节点表示能够在不同的局部邻域范围内捕获结构信息。通过引入注意力机制，可以自适应地为不同层的表示分配权重，并使用跳跃知识结合策略在不同的层次之间传递共享信息。

3) 通过公开的以太坊数据集构建实验数据集进行地址分类实验，实验结果表明，所提模型在有效性、准确度等方面均优于其他基线方法。

1 相关工作

1.1 区块链交易地址分类技术

现有的针对以太坊的去匿名化研究仍然是以地址分类和地址聚类为主。其研究主要集中于人工特征工程结合使用机器学习方法、图建模交易分析等方面。文献[9]提出了一种新的基于机器学习的方法，用于在以太坊区块链上检测非法账户。该方法使用了多种特征，包括交易数量、交易额、交易频率等，以便对账户进行分类。文献[10]提出了使用长短期记忆（LSTM, long short-term memory）网络分类和检测以太坊智能合约的方法。将智能合约的事务序列分为交易和合约创建两类，并根据事务的数量、输入和输出的价值等特征来对它们进行分类和检测。前期采用的人工特征工程依赖于设计人员的先验知识，且无法捕获区块链数据中的交易模式信息，导致特征利用率低，表达性不佳。因此，有研究人员将以太坊交易建模为图或网络，地址为节点，交易为边，使用网络嵌入技术来自动捕获网络中的账户交互特征，进而应用于下游的分类和聚类任务。文献[11]设计了一种名为 Trans2Vec 的网络嵌入算法来提取以太坊地址的特征，用于后续的钓鱼识别。文献[12]提出了一种用于以太坊网络钓鱼账户检测的级联特征提取方法，该方法不仅可以提取利用账户的特征，还可以提取利用其邻居的特征。文献[13]将以太坊交易网络表示为一个加权有向图，采用子图机制，通过图卷积层结合图自编码器无监督的方式进行图嵌入，并利用 LightGBM[14]实现了钓鱼账户的分类。文献[15]使用 Node2Vec 算法提取账户的潜在特征，并通过支持向量机（SVM, support vector machine）对钓鱼者进行分类。文献[16]提出了一种名为 BlockGC 的联合学习框架，用于在

区块链上对账户身份进行推断。该框架使用图卷积网络来提取特征，并使用对比学习优化学习过程，最小化同一账户的不同表示之间的距离，从而提高推断准确性。文献[17]通过在图神经网络中引入自适应的节点表示学习来增强账户分类性能，并使用交易量、交易频率等特征来过滤无关账户。但上述方法并未对交易网络中重要中间地址节点赋予足够的关注度，在模型识别效果上存在改进空间。

1.2　图神经网络及其变种在分类中的研究

图神经网络是一种用于图数据分析和挖掘的深度学习模型。与传统的深度学习模型不同，图神经网络不仅可以处理欧几里得空间中的数据，也可以处理非欧几里得空间中的数据，比如图结构数据、社交网络、化学分子结构等。

图神经网络的基本原理是将节点和边的特征向量进行消息传递和聚合，最终输出每个节点的表示向量，用于节点分类、图分类、链接预测等任务。GNN 主要分为两类，即基于图卷积神经网络（GCN, graph convolutional network）[18]的传统方法和基于消息传递机制的新方法。

基于 GCN 的传统方法主要利用矩阵卷积运算来处理节点和边的特征向量，如 ChebNet、GCN 等，这种方法的优点在于可以利用图结构的局部邻域信息更新节点特征向量，但是不能处理高维特征和动态图结构。

基于消息传递机制的新方法主要利用节点的邻居信息来更新节点的特征向量，如 GraphSAGE、GAT 等。这种方法的优点在于可以处理高维特征和动态图结构。

GraphSAGE[19]是一种基于图神经网络的节点表示学习算法，旨在对大规模图进行有效的节点表示学习。与其他 GNN 模型一样，GraphSAGE 基于邻居信息对节点进行聚合，从而学习到每个节点的向量表示，在这个过程中，GraphSAGE 学习到的是聚合方式而非特定顶点的固定嵌入向量。模型只需要输入节点特征集合及其采样的邻居节点集合即可，不需要将整张图输入模型。因此在学习到合适的聚合方式后，可将其快速应用到新的图结构。

GraphSAGE 的核心思想是聚合邻居信息来计算节点表示，其具体流程如下。

1) 邻居采样。对于每个节点，从其邻居节点中随机采样一定数量的节点，并将其放入集合中。

2) 聚合邻居信息。对于每个节点，将其本身的特征向量与邻居节点的特征向量进行聚合，得到节点的向量表示。这里使用的是一个多层感知器（MLP, multi-layer perceptron）模型，其参数可以在训练过程中进行学习。具体的聚合方式如式(1)所示。

$$\boldsymbol{h}_v^k = \sigma(\boldsymbol{W}^k \mathrm{CONCAT}(\boldsymbol{h}_v^{k-1}, \mathrm{AGG}(\{\boldsymbol{h}_u^{k-1}\}_{u\in N(v)}))) \tag{1}$$

其中，AGG 表示聚合运算，$k \in 1,2,\cdots,K$ 表示层数，$\boldsymbol{h}_v^k$ 表示第 k 层的节点，$\boldsymbol{W}^k$ 表示第 k 层的权重矩阵，CONCAT 表示将多个向量进行拼接，σ 表示激活函数，$\boldsymbol{h}_u^{k-1}$ 表示第 $k-1$ 层的邻居节点。

3) 聚合结果的汇总。对于每个节点 v，将其第 k 层的向量表示 $\boldsymbol{h}_v^k$ 汇总作为该节点的最终表示。常用的聚合函数有最大池化、平均池化、LSTM 等。例如平均池化，即将所有 k 层的向量表示求平均。

与其他 GNN 模型不同，GraphSAGE 不需要使用所有节点的邻居信息来更新节点表示，而是使用采样的方式，从而大大减少了计算复杂度。同时，GraphSAGE 可以灵活地使用不同的层数和聚合方法，从而适于不同的任务。

2　系统模型

2.1　模型框架

AJKGS-ABCM（attention jumping knowledge graph SAGE account-based blockchain classification model）是一种基于图神经网络的多分类模型，用于以太坊地址多分类问题，其模型框架如图 1 所示。通过 AJK-GraphSAGE 算法对邻居节点赋予权重和融合不同层的特征向量，充分利用了深度神经网络的特点和信息融合的能力，提升了模型的表达能力和分类精度。经过对比实验验证，该模型在账户余额模型区块链地址分类任务中表现出较好的性能，但值得注意的是，该方法目前无法解决虚拟币混淆场景的细粒度地址分类，只能确定其大类信息。以下是该模型框架各部分的详细描述。

1) 输入层：以太坊地址的原始特征作为输入节点表示。

2) GraphSAGE 层：包括 K 个 GraphSAGE 层，每层都负责从邻居节点聚合信息以学习输入节点的新表示。

3) 注意力权重计算层：将节点在不同 GraphSAGE 层的表示连接起来，计算注意力权重。

4) 跳跃知识结合策略层：利用注意力权重将不

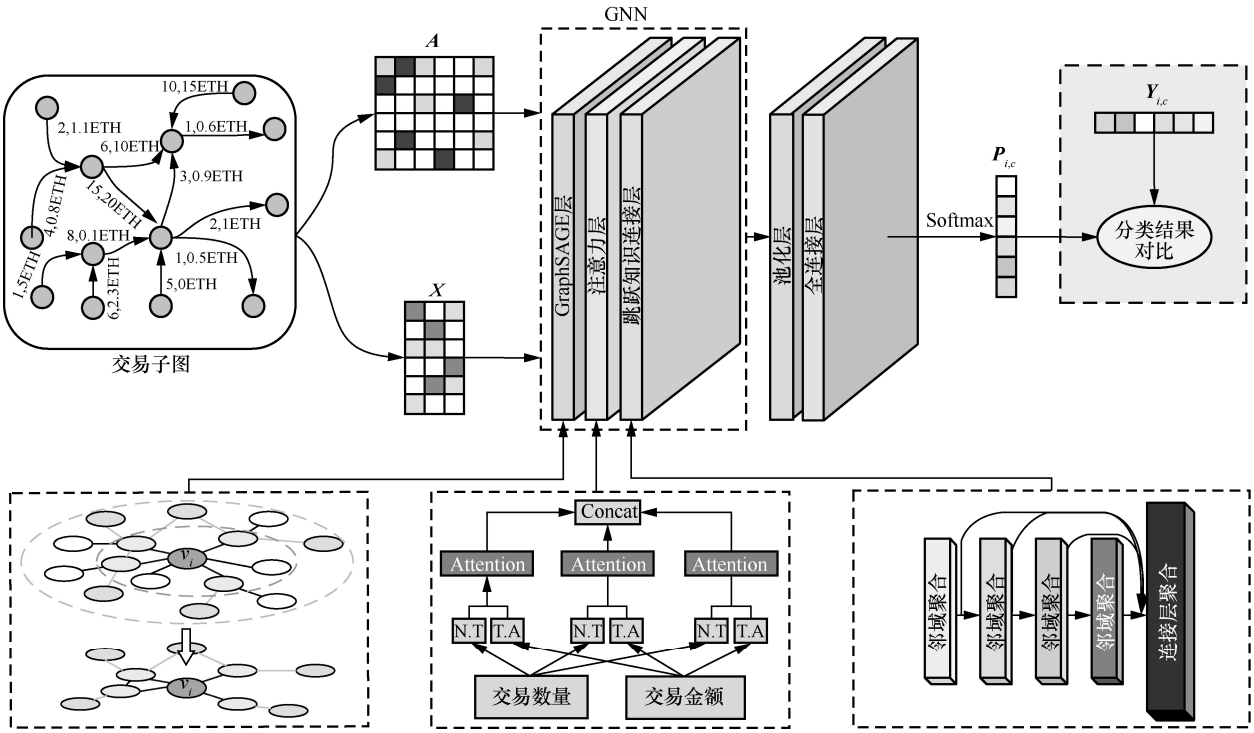


图 1 AJKGS-ABCM 模型框架

同 GraphSAGE 层节点表示组合形成最终节点表示。

5) 全连接分类层：将最终节点表示映射到类别概率空间。

6) Softmax 层：应用 Softmax 激活函数以获取每个节点属于每个类别的概率。

7) 交叉熵损失计算层：计算模型预测的类别概率与真实类别标签之间的交叉熵损失。

8) 随机梯度下降（SGD, stochastic gradient descent）优化器：使用随机梯度下降算法优化模型参数以最小化损失。

2.2 符号定义

本文所涉及的参数含义如表 1 所示。

表 1 参数含义

参数	含义
N	节点的数量
$\boldsymbol{A}$	图的邻接矩阵，其中 A_{ij} 表示节点 i 和节点 j 之间是否有边
$\boldsymbol{X}$	节点特征矩阵，其中 $\boldsymbol{x}_i$ 表示节点 i 的特征向量
k	GraphSAGE 的层数，$k=1,2,\cdots,K$，K 为总层数
C	分类类别的数量，其中 c=1,2,…,C 表示每个类别
Θ	模型参数集合，包括 GraphSAGE 层的权重矩阵 $\boldsymbol{W}^1,\boldsymbol{W}^2,\cdots,\boldsymbol{W}^K$ 、注意力权重矩阵 $\boldsymbol{W}_a$ 和分类层权重矩阵 $\boldsymbol{W}_c$
η	学习率，用于随机梯度下降算法
$\boldsymbol{h}_i^k$	节点 v_i 在第 k 层的表示
$\mathcal{N}(v_i)$	节点 v_i 的邻居节点集合
$\boldsymbol{H}^1,\boldsymbol{H}^2,\cdots,\boldsymbol{H}^K$	经过 K 个 GraphSAGE 层的输出表示矩阵
$\boldsymbol{A}_i^k$	节点 v_i 的注意力权重向量，表示第 k 层 GraphSAGE 层输出表示的重要性
$\boldsymbol{H}_i^{\text{final}}$	节点 v_i 经过注意力和跳跃知识结合策略处理后的最终表示
O	分类层的输出
$\boldsymbol{P}$	类别概率矩阵，其元素 $P_{i,c}$ 表示节点 v_i 属于类别 c 的概率
$\boldsymbol{Y}$	节点的真实类别标签矩阵，表示每个节点的独热编码类别标签
L	交叉熵损失函数数值，用于衡量模型预测与真实类别标签之间的一致性
$\nabla_\Theta L$	损失函数关于模型参数的梯度

3　模型设计

3.1　输入层及 GraphSAGE 层设计

首先，需要输入以太坊地址节点特征矩阵、邻接矩阵以及权重矩阵，并初始化节点表示。

输入特征矩阵。对于图 $G=(V,E)$，需要一个输入特征矩阵 $\boldsymbol{X}\in\mathbb{R}^{N\times F_0}$，其中 N 表示节点数量，F_0 表示输入特征的维度。矩阵 $\boldsymbol{X}$ 中的每一行 $\boldsymbol{x}_i$ 表示节点 v_i 的特征向量。

输入邻接矩阵。$\boldsymbol{A}$ 为图的邻接矩阵，其中，节点 i 与节点 j 有边连接则 A_{ij}=1，否则 A_{ij}=0。

输入权重矩阵。$\boldsymbol{W}^1,\boldsymbol{W}^2,\cdots,\boldsymbol{W}^K$ 为交易子图每一层权重矩阵，由交易次数、交易金额计算并通过训练更新。交易子图中的数字和字母含义为（交易次数，交易金额）。

初始化节点表示。将输入特征矩阵 $\boldsymbol{X}$ 作为节点表示的初始值。对于每个节点 v_i，其初始表示为 $\boldsymbol{h}_i^0=\boldsymbol{x}_i$。将所有节点的表示组合成一个矩阵 $\boldsymbol{H}^0\in\mathbb{R}^{N\times F_0}$。其中，$\boldsymbol{H}^0$ 中的每一行 $\boldsymbol{h}_i^0$ 表示节点 v_i 的初始表示，即特征矩阵 $\boldsymbol{X}$ 就是 $\boldsymbol{H}^0$。如图 2 所示，$\boldsymbol{A}$ 为邻接矩阵，$\boldsymbol{X}$ 为特征矩阵。

首先，使用输入特征矩阵初始化节点表示。接下来，将使用这些表示作为 GraphSAGE 层的输入，以更新节点表示并捕获图中的结构信息。

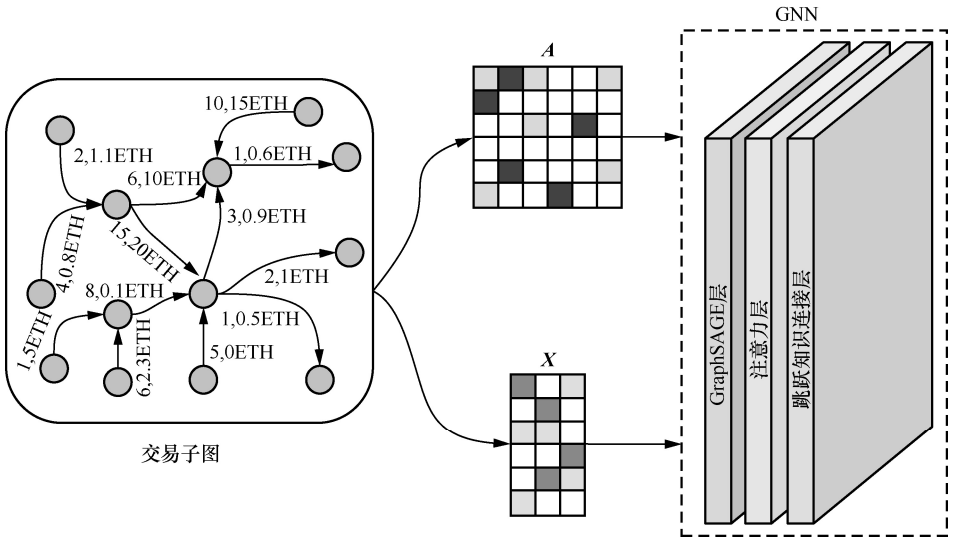


图 2　初始化邻接矩阵与特征矩阵

接着，执行 GraphSAGE 层的计算，以下是详细过程描述，如图 3 所示。

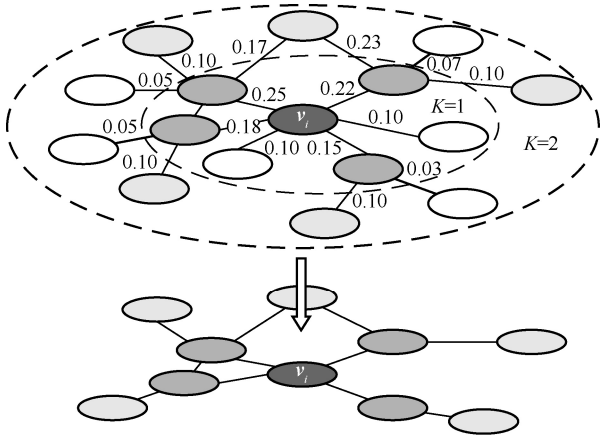


图 3　GraphSAGE 层采样聚合

1) 邻居节点的表示聚合。对于每个节点 v_i，需要聚合其邻居节点的表示。在 k 层，节点 v_i 的邻居节点的表示聚合为

$$\text{AGGREGATE}^k(\boldsymbol{h}_i^k,\mathcal{N}(v_i))=\frac{1}{|\mathcal{N}(v_i)|}\sum_{v_j\in\mathcal{N}(v_i)}\boldsymbol{W}^k\boldsymbol{h}_j^k \quad (2)$$

其中，AGGREGATE 表示聚合 $\mathcal{N}(v_i)$ 表示节点 v_i 的邻居集合，$\boldsymbol{W}^k\in\mathbb{R}^{F_k\times F_{k+1}}$ 表示第 k 层的权重矩阵。

2) 更新节点表示。将节点的表示与其邻居表示聚合相结合以生成更新后的表示，如式(3)所示。

$$\boldsymbol{h}_i^{k+1}=\sigma(\boldsymbol{W}^k\boldsymbol{h}_i^k+\text{AGGREGATE}^k(\boldsymbol{h}_i^k,\mathcal{N}(v_i))) \quad (3)$$

其中，σ 为 ReLU 激活函数。

在这个阶段，通过执行 GraphSAGE 层计算来更新节点表示。这些表示将捕获图中的结构信息。接下来，引入注意力机制和跳跃知识结合策略来进一步优化这些表示。

3.2　AJK-GraphSAGE 算法设计

相较于 GraphSAGE 算法，AJK-GraphSAGE 算

法引入了注意力机制和跳跃知识结合策略优化节点表示。在 GraphSAGE 算法中，每个节点的邻居节点特征用平均池化来生成节点的新特征，这可能会损失一些关键信息，因此，通过引入注意力机制给每个邻居节点赋予不同的注意力权重，将更多的注意力集中在重要的邻居节点上，从而更好地表达节点的特征。跳跃知识结合策略将节点在前面多个聚合层中学习到的邻居节点特征向量进行跳跃连接，得到节点的跳跃聚合特征向量。然后将节点的一阶邻居特征向量和跳跃聚合特征向量进行拼接得到最终的特征向量。该算法充分利用之前层次的信息，提高了模型的表达能力。详细过程描述如下。

该算法的输入数据为经过 K 个 GraphSAGE 层的输出表示矩阵 $\boldsymbol{H}^1,\boldsymbol{H}^2,\cdots,\boldsymbol{H}^K$。

首先，计算注意力权重，注意力权重可以表示不同邻居节点表示之间的重要性。对于每个节点 v_i，计算一个注意力权重向量 $\boldsymbol{A}_i$，表示不同 GraphSAGE 层输出表示的重要性，其权重主要参考以太坊地址间的交易次数及交易金额。

$$\boldsymbol{A}_i=\mathrm{softmax}\left(\boldsymbol{W}_a\mathrm{CONCAT}\left(\boldsymbol{h}_i^1,\boldsymbol{h}_i^2,\cdots,\boldsymbol{h}_i^K\right)\right) \tag{4}$$

其中，$\boldsymbol{W}_a\in\mathbb{R}^{(F_1+F_2+\cdots+F_K)\times K}$ 是注意力权重矩阵，$\boldsymbol{h}_i^k$ 是节点 v_i 在第 k 层的表示。

接着，应用跳跃知识结合策略，在不同层之间传递信息，从而捕获不同层的邻域结构以达到更好的节点表示，如图 4 所示。

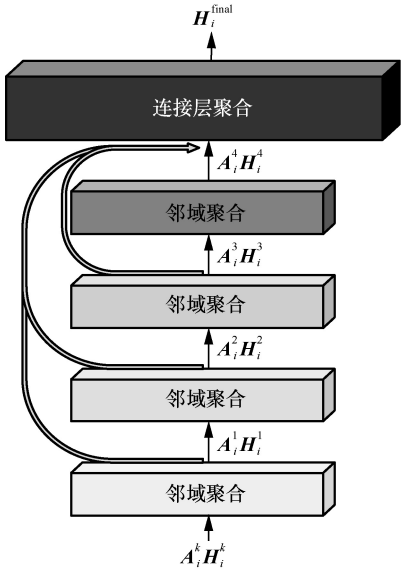


图 4 4 层跳跃知识结合

下面使用计算得到的注意力权重 $\boldsymbol{A}_i$ 来组合来自不同 GraphSAGE 层的节点表示

$$\boldsymbol{H}_i^{\text{final}}=\sum_{i=1}^{K}\boldsymbol{A}_i^k\boldsymbol{H}_i^k \tag{5}$$

其中，$\boldsymbol{A}_i^k$ 表示节点 v_i 的第 k 个注意力权重。经过这一步骤，可以得到经过注意力和跳跃知识结合策略处理后的节点表示矩阵 $\boldsymbol{H}^{\text{final}}\in\mathbb{R}^{N\times F_{\text{final}}}$。

在此算法中，通过引入注意力机制和跳跃知识结合策略优化节点表示，可以更好地提取以太坊交易图中的结构信息，进行后续以太坊地址多分类任务。

3.3 全连接层及损失函数设计

在这一阶段中，首先，将经过注意力机制和跳跃知识结合策略处理后的节点表示用全连接层映射到以太坊地址多分类任务的类别概率空间，全连接层的输出为

$$O=\boldsymbol{H}^{\text{final}}\boldsymbol{W}_c \tag{6}$$

其中，$\boldsymbol{W}_c\in\mathbb{R}^{F_{\text{final}}\times C}$ 是分类层的权重矩阵，C 是类别数量。

全连接层的输出随后通过应用 Softmax 激活函数来获得最终的类别概率。

$$\boldsymbol{P}=\mathrm{Softmax}(O) \tag{7}$$

经过上述操作，可以得到类别概率矩阵 $\boldsymbol{P}\in\mathbb{R}^{N\times C}$，其元素 $P_{i,c}$ 表示节点 v_i 属于类别 c 的概率。

然后，通过计算交叉熵损失，可以衡量模型预测与真实类别标签之间的一致性。损失函数将用于优化模型参数。在该阶段中，需要输入节点的真实类别标签矩阵 $\boldsymbol{Y}\in 0,1^{N\times C}$。对于每个节点 v_i 和每个类别 c，计算损失函数 L 为

$$L=-\frac{1}{N}\sum_{i=1}^{N}\sum_{c=1}^{C}Y_{i,c}\log(P_{i,c}) \tag{8}$$

其中，N 表示节点数量，C 表示类别数量，$Y_{i,c}$ 表示节点 v_i 的真实类别标签的独热编码，$P_{i,c}$ 表示节点 v_i 属于类别 c 的预测概率。

最后，通过使用随机梯度下降算法最小化损失函数值，从而优化模型参数以提高预测性能。在这一步骤中，使用损失函数值 L 以及模型参数集合 $\Theta=\boldsymbol{W}^1,\boldsymbol{W}^2,\cdots,\boldsymbol{W}^K,\boldsymbol{W}_a,\boldsymbol{W}_c$ 来计算损失函数关于模型参数的梯度

$$\nabla_\Theta L=\frac{\partial L}{\partial\Theta} \tag{9}$$

然后，使用随机梯度下降更新模型参数

$$\Theta_{t+1}=\Theta_t-\eta\nabla_\Theta L_t \tag{10}$$

其中，η 表示学习率，Θ_t 和 Θ_{t+1} 分别表示在第 t 次和第 $t+1$ 次迭代时的模型参数。

4　实验评估

本节将对基于 AJK-GraphSAGE 的以太坊交易地址分类模型 AJKGS-ABCM 进行相关实验验证，评价指标采用精确率（Precision）、召回率（Recall）、F1 分数（F1-score）、微观 F1（micro-F1）、宏观 F1（macro-F1）等。同时，为了分析本文方法的效果，选取了多种同类方法进行对比实验，以验证分类效率。

4.1　实验设计及参数描述

1) 数据集描述

本文使用的标签数据集整合自 Etherscan.io、Kaggle 以太坊标签数据集、XLabelCloud 数据集以及其他通过社交软件获取的以太坊地址标签。通过对上述数据集整合去重，并去除交易值为零的无效账户，共获得带标签以太坊地址 22 312 个，地址标签包含交易所、矿池/矿工等 11 个分类标签，如表 2 所示。上述以太坊标签数据集仅包含相应以太坊地址被查询时账户状态信息，包含地址余额、交易次数等信息，部分地址数据集仅带有地址标签，为获取更多账户特征信息来构建以太坊交易网络，可通过以太坊浏览器 Etherscan API 来获取数据，包含区块、交易 hash 和交易 from 地址和交易 to 地址等信息。但因为以太坊浏览器都设有爬虫限制，为了获取更多的以太坊交易数据，本文还结合了 Google BigQuery 平台的 crypto_ethereum 交易数据集。

本文通过上述方法获取了 22 312 个带标签以太坊地址的所有交易记录，共计 3 437 967 条交易数据，但因为部分地址交易数据具有极大的稀疏性，且为了构建相对完整的以太坊交易网络，本文对上述交易数据中所涉及的地址进行二次交易数据获取，共获取 58 172 811 条交易数据，对上述共 61 610 778 条数据去除 isError 字段为 1 的数据，共计剩余交易数据 59 465 095 条，并将其处理为节点邻接矩阵。

2) 节点特征构造

上文通过遍历交易数据生成了所有地址的邻接矩阵文件作为模型训练的输入之一，但要想充分学习以太坊交易特征，除了构造交易图结构外，还需要一定的节点特征信息。以太坊账户在状态特征上类似于传统点对点支付账户，账户状态时刻变化，通过 API 调用只能获取特定账户诸如账户余额等最新状态信息，对构造节点特征矩阵的意义较小。因此同样通过遍历上述交易数据，为每个节点构造建议的特征矩阵。

遍历获得以下节点特征字段：交易总金额 total_value、交易最小金额 min_value、交易最大金额 max_value、发送总金额 total_value_sent、接收总金额 total_value_received、当前账户余额 current_bal-ance、交易总次数 total_transactions、交易最短间隔 min_interval、交易最长间隔 max_interval、交易频率 frequency、发送交易次数 sent_transactions、接收交易次数 received_transactions、创建合约次数 cre-ated_contracts、调用合约次数 called_contracts、最

表 2　以太坊数据集标签类型

标签名称	标签符号	标签描述
交易所	Exchange	该地址属于加密货币交易所，可能涉及加密货币的买卖交易和其他金融服务
矿池/矿工	Mining_pool/Mining	该地址是由一个或多个矿工汇集成的矿池地址，可能涉及加密货币挖矿等活动
博彩	Gambling	该地址可能是在线博彩网站的地址，可能涉及加密货币博彩活动
合约代币	Token	该地址属于一个以太坊代币合约，可能用于代币发行、代币交易等活动
暗网交易	Darknet	该地址可能是一个用于在暗网进行加密货币交易的地址，可能涉及非法交易
钱包	Wallet	该地址可能属于一个以太坊钱包，可能用于存储、管理和转移加密货币
欺诈	Scam	该地址可能是用于进行欺诈活动的地址，可能涉及虚假交易、社工钓鱼、伪造代币等活动
筹资活动	Ico	该地址可能是用于 Ico 或其他筹资活动的地址，可能涉及投资、筹款等活动
黑客	Hacker	该地址可能是用于黑客盗币活动的地址，可能涉及攻击以太坊网络、其他区块链网络或其他网络的活动
货币清洗	Mixer	该地址可能是用于混合加密货币交易的地址，可能涉及加密货币的匿名化处理
普通用户	User	该地址属于一般的以太坊用户，可能仅用于普通的加密货币交易等活动

常接收交易的地址 most_common_receiver、最常发送交易的地址 most_common_sender、交易总手续费 total_gas_used、交易平均手续费 average_gas_used、交易失败次数 total_failed_transactions。

3) 实验评估指标及参数设置

为了评估所提出的基于 AJK-GraphSAGE 的以太坊交易地址分类模型 AJKGS-ABCM 在实际分类实验中的效率，本文使用了以下指标，包括 Precision、Recall、F1-score、micro-F1、macro-F1，详细信息如表 3 所示。以交易所地址标签为例，在本文中，真阳性（TP）代表被分类模型正确分类为交易所地址的数量；假阳性（FP）代表被分类模型识别为交易所地址的其他分类标签数量；真阴性（TN）代表被分类模型正确分类为其他分类标签的数量；假阴性（FN）代表被分类模型识别为其他分类标签的交易所地址的数量。

本文中所有实验均在 Ubuntu 18.1 64 位操作系统上进行，其中 CPU 为 32 核心 AMD EPYC 75F3，GPU 为 NVIDIA GeForce RTX 3090，8T 三星 M.2 SSD 固态硬盘，416 TB 7200RPM SATA，所有模型都在 Pytorch1.10.2 环境中使用 Adgrad 优化器实现，Python 版本为 3.8。为了与同类型分类方法相比较，本文使用了人工特征结合 KNN[20]、DeepWalk[21]、Node2Vec[22]、Struc2Vec[23]、I2BGNN[19]、GCN[18]、GraphSAGE[19]等方法进行分类实验。对于模型参数，本文模型学习率设置为 0.01，batch-size 大小设置为 64，激活函数设置为 ReLU，dropout 设置为 0.05，采样层的数量设置为 2，每个采样层的邻域采样大小设置为 20，其余参数与其他基线模型一致，同时所有基于图嵌入方法的嵌入或输出维数都设置为 128，epoch 设置为 200。对于基于随机步行的嵌入方法，节点嵌入维数设置为 128，窗口大小设置为 4，步行长度设置为 20，每个节点步行设置为 4。对于 Node2Vec，设置 p 为 0.25，q 为 0.4。GraphSAGE 采用 mean 聚合，其他参数与上述一致。为了进行综合对比，随机选取带标签目标地址中的 {60%,70%,80%} 作为训练集，其余带标签地址作为测试集，所有实验重复进行 20 次取平均值进行实验效果对比，详细实验结果如表 4～表 6 所示，其中，Exchange、Mining 等表示地址类型，平均实验结果如表 7 所示。

表 3　　实验评估指标

指标	计算式	含义
Precision	$P=\dfrac{\mathrm{TP}}{\mathrm{TP+FP}}$	被正确分类为某标签的地址占总的被分类为某标签的地址的比例
Recall	$R(\mathrm{TPR})=\dfrac{\mathrm{TP}}{\mathrm{TP+FN}}$	所有正例中被正确分类的比例
F1-score	$\mathrm{F1\text{-}score}=2\dfrac{PR}{P+R}$	兼顾 Precision 和 Recall 的指标
micro-F1	$\mathrm{micro\text{-}F1}=\dfrac{\sum_{i=1}^{C}2\mathrm{TP}_i}{\sum_{i=1}^{C}2\mathrm{TP}_i+\mathrm{FP}_i+\mathrm{FN}_i}$	先计算所有类别的总的 Precision 和 Recall，后计算 F1 值，即 micro-F1
macro-F1	$\mathrm{macro\text{-}F1}=\dfrac{1}{C}\sum_{i=1}^{C}\dfrac{2\mathrm{TP}_i}{2\mathrm{TP}_i+\mathrm{FP}_i+\mathrm{FN}_i}$	将所有类别的 Precision 和 Recall 求平均，然后计算 F1 值，即 macro-F1

表 4　　以太坊多分类各方法 Precision 对比

方法	Exchange	Mining	Gambling	Token	Darknet	Wallet	Scam	Ico	Hacker	Mixer	User
KNN	0.683	0.675	0.651	0.643	0.648	0.656	0.651	0.624	0.638	0.671	0.661
Deepwalk	0.722	0.715	0.708	0.707	0.701	0.725	0.695	0.727	0.711	0.733	0.728
Node2Vec	0.735	0.731	0.715	0.722	0.719	0.731	0.718	0.734	0.724	0.740	0.723
Struc2Vec	0.741	0.735	0.724	0.731	0.730	0.745	0.729	0.738	0.733	0.744	0.739
I2BGNN	0.831	0.812	0.819	0.826	0.822	0.842	0.833	0.840	0.832	0.829	0.841
GCN	0.821	0.812	0.810	0.809	0.791	0.822	0.811	0.819	0.801	0.820	0.815
GraphSAGE	0.843	0.821	0.819	0.826	0.815	0.833	0.825	0.838	0.841	0.845	0.835
AJKGS-ABCM	0.875	0.856	0.845	0.861	0.855	0.871	0.854	0.870	0.873	0.881	0.859

表 5　以太坊多分类各方法 Recall 对比

方法	Exchange	Mining	Gambling	Token	Darknet	Wallet	Scam	Ico	Hacker	Mixer	User
KNN	0.665	0.661	0.642	0.635	0.638	0.649	0.638	0.619	0.626	0.655	0.648
Deepwalk	0.715	0.720	0.695	0.698	0.688	0.714	0.681	0.709	0.701	0.716	0.715
Node2Vec	0.712	0.720	0.705	0.709	0.706	0.722	0.703	0.718	0.709	0.729	0.716
Struc2Vec	0.733	0.718	0.715	0.722	0.719	0.737	0.721	0.726	0.712	0.729	0.715
I2BGNN	0.823	0.809	0.811	0.815	0.806	0.818	0.825	0.836	0.814	0.811	0.829
GCN	0.811	0.806	0.801	0.795	0.785	0.807	0.803	0.812	0.795	0.808	0.802
GraphSAGE	0.831	0.809	0.812	0.811	0.803	0.818	0.814	0.818	0.825	0.831	0.817
AJKGS-ABCM	0.855	0.841	0.838	0.852	0.836	0.862	0.846	0.857	0.861	0.870	0.849

表 6　以太坊多分类各方法 F1-score 对比

方法	Exchange	Mining	Gambling	Token	Darknet	Wallet	Scam	Ico	Hacker	Mixer	User
KNN	0.674	0.668	0.646	0.639	0.643	0.652	0.644	0.622	0.632	0.663	0.654
Deepwalk	0.718	0.710	0.702	0.703	0.694	0.719	0.688	0.718	0.706	0.724	0.721
Node2Vec	0.723	0.725	0.710	0.716	0.712	0.726	0.710	0.726	0.716	0.734	0.719
Struc2Vec	0.737	0.726	0.719	0.726	0.724	0.741	0.725	0.732	0.722	0.736	0.727
I2BGNN	0.827	0.816	0.815	0.820	0.814	0.830	0.829	0.838	0.823	0.820	0.835
GCN	0.816	0.809	0.805	0.802	0.788	0.814	0.807	0.816	0.798	0.814	0.808
GraphSAGE	0.837	0.815	0.816	0.818	0.825	0.825	0.819	0.828	0.833	0.838	0.826
AJKGS-ABCM	0.865	0.849	0.841	0.856	0.866	0.866	0.850	0.864	0.867	0.875	0.854

表 7　以太坊多分类各方法平均实验结果对比

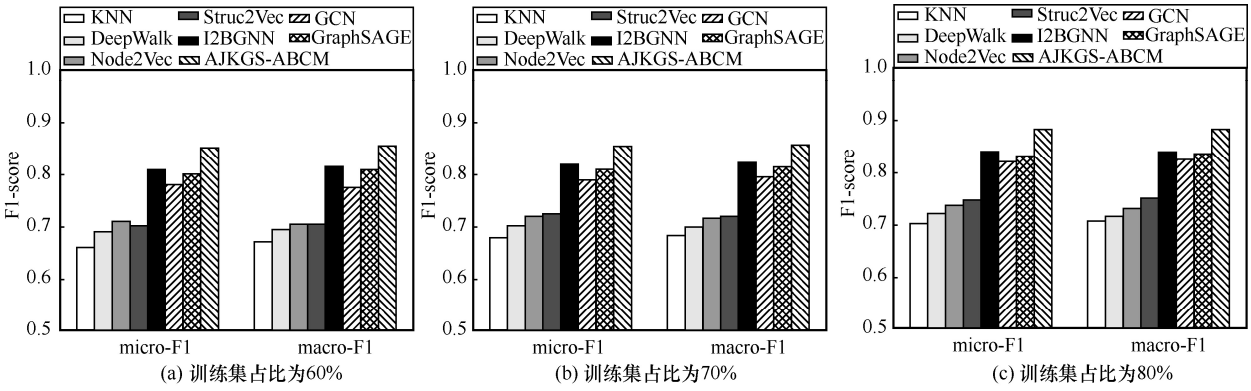
方法	Precision	Recall	micro-F1	macro-F1
KNN	0.655	0.643	0.645	0.648
DeepWalk	0.716	0.703	0.703	0.709
Node2Vec	0.725	0.713	0.715	0.719
Struc2Vec	0.734	0.722	0.725	0.727
I2BGNN	0.83	0.819	0.825	0.824
GCN	0.811	0.802	0.801	0.807
GraphSAGE	0.828	0.819	0.818	0.823
AJKGS-ABCM	0.862	0.852	0.849	0.856

4.2　实验结果分析

通过图 5 可以看出，在相同实验参数下，不同比例的训练集对各方法的分类结果存在一定的影响，随着训练集占比的增大，各方法的分类效果均得到了一定程度的提高。因此，本文后续的实验对比中将训练集与测试集的占比固定为 8:2，以此获得较好的实验结果。

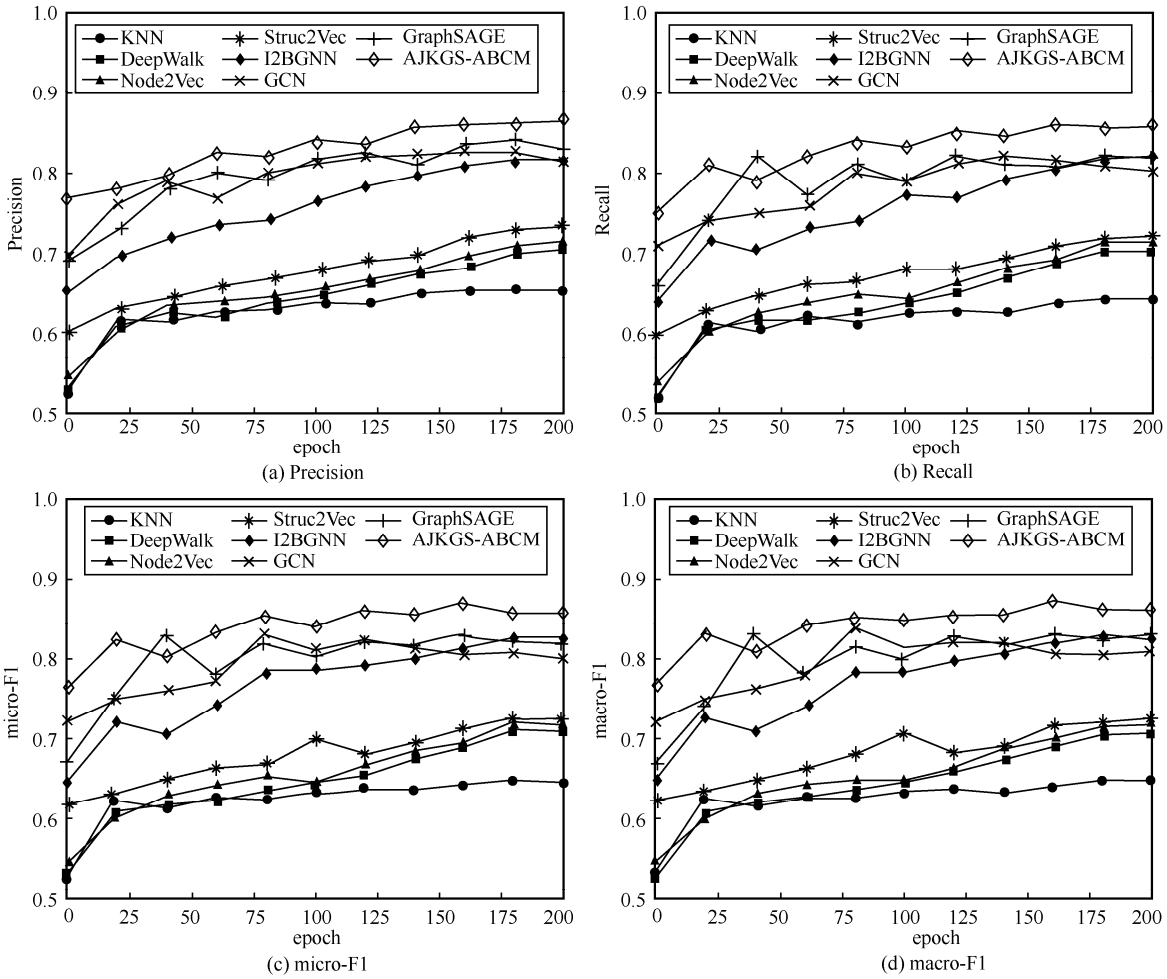
同时，由图 6 可知，本文提出的 AJKGS-ABCM 模型的检测精度整体上优于其他基线模型，并且可以较快地达到收敛状态。在所有方法中，相较于其他基于图嵌入的方法，基于手工特征结合 KNN 的方法效果最差，这是因为传统的机器学习方法难以捕捉区块链交易网络中的拓扑信息，仅使用账户地址本上的属性特征导致分类精度不理想。

Struc2Vec、DeepWalk 以及 Node2Vec 这 3 种方法的分类效果相近，其中 Struc2Vec 优于其他 2 种方法，在 200 epoch 下精确率分别提高了 2.82%、1.91%。I2BGNN、GCN、GraphSAGE 以及 AJKGS-ABCM 都是基于图神经网络的方法，4 种方法均可以充分利用地址本身特征并结合利用邻居特征，AJKGS-ABCM 方法相较于其他 3 种方法，在 200 epoch 下精确率分别提高了 4.35%、4.84%、3.39%；召回率分别提升了 4.71%、5.45%、4.78%；微观 F1 分别提高了 2.82%、5.26%、3.53%；宏观 F1 分别提高了 2.98%、5.40%、3.66%。这主要是因为 AJKGS-ABCM 引入了注意力机制及跳跃知识结合策略，能够充分利用邻居特征信息，融合不同层的特征向量，使模型在不同的抽象层次上学习到更有用的信息，进而提高了模型的预测性能。



(a) 训练集占比为60%　(b) 训练集占比为70%　(c) 训练集占比为80%

图 5　不同比例训练集划分下各方法的实验结果



(a) Precision　(b) Recall　(c) micro-F1　(d) macro-F1

图 6　不同 epoch 下各方法的性能

表 4 和表 5 分别为各方法在 200 epoch 下对以太坊地址多分类任务在数据集中各标签的分类精确率和召回率。从表 4 和表 5 中可以看到，所有方法对 Exchange、Mining、Ico 以及 Mixer 的识别率较高，识别精确率和召回率明显高于其他地址，由此可见上述 4 类地址特征信息较明显，相较于其他类地址能够得到清晰的分类。同时可以发现，AJKGS-ABCM 在大多数分类下的精确率和召回率最高，其次是 I2BGNN 和 GraphSAGE，这 3 种方法的表现相对较好。而 KNN 的表现相对较差，特别是在 Ico 分类下，其精确率只有 62.5%，召回率只有 61.9%，远低于其他方法。从表 6 整体上看，AJKGS-ABCM 在所有类上都表现出最高的 F1-score，表明该方法在以太坊多分类问题上具有

最佳性能。其次是 I2BGNN、GraphSAGE 和 GCN 方法，这 3 种方法的表现相对较好。相比之下，KNN 性能相对较差，可能是因为其作为一种机器学习方法，在复杂的图结构数据上不能很好地捕捉到潜在的特征信息。

从 KNN 到 DeepWalk 有显著的性能提升。这表明引入图嵌入技术（如 DeepWalk）可以在一定程度上改善以太坊多分类任务的性能。Node2Vec 和 Struc2Vec 与 DeepWalk 相比有所提高，说明不同的网络表示的学习方法对性能有影响。I2BGNN、GCN、GraphSAGE 以及 AJKGS-ABCM 都是基于图神经网络的方法，它们的性能明显优于基于图嵌入的方法（如 DeepWalk、Node2Vec 和 Struc2Vec），这表明 GNN 方法在以太坊多分类任务中更具优势。

在所有方法中，可以观察到 Mixer 和 User 的 F1 值较高，而 Gambling、Token 和 Darknet 的 F1 值较低。这可能是因为不同分类的特征差异较大，某些类的特征更容易被模型捕捉，导致模型在不同分类下的表现存在较大差异。

通过表 7 可以看到，在整体性能上，AJKGS-ABCM 在 Precision、Recall、micro-F1 和 macro-F1 上均表现出最高的得分，证明其在以太坊多分类任务上具有最佳性能。相比之下，KNN 的性能较差。AJKGS-ABCM 在 Precision 和 Recall 上都有较高的得分，表明该方法可以在保持高查准率的同时，实现高查全率。这也意味着该方法在正确检测到相关类的同时，不会产生过多的误报。同时可以看到，所有基于图神经网络的方法的性能明显优于其他基于图嵌入的方法（如 DeepWalk、Node2Vec 和 Struc2Vec）或基于机器学习的方法（如 KNN），这表明基于图神经网络的方法在以太坊多分类任务中更容易取得较好的分类效果。除此之外，从各个分析指标来看，AJKGS-ABCM 的 Precision 最高，为 86.2%；其次是 I2BGNN，为 83.0%；而 KNN 方法的 Precision 最低，为 65.5%。这表明 AJKGS-ABCM 方法在预测分类时更准确。

5　结束语

本文提出了一种基于图神经网络的账户余额模型区块链地址分类模型 AJKGS-ABCM。从以太坊的交易数据出发，将其建模为图，把以太坊地址作为节点，以太坊地址间交易作为边。接着运用本文提出的 AJK-GraphSAGE 方法学习图的嵌入表示，将节点及其采样的邻居节点集合作为模型的输入。该方法能够自适应地聚合邻居节点的信息，使节点表示能够在不同的局部邻域范围内捕获到结构信息。通过引入注意力机制，可以自适应地为不同层的表示分配权重。使用跳跃知识结合策略在不同的层次之间传递共享信息，提高了训练速度和泛化能力。最后进行了实验对比，验证了方法模型的准确度和有效性。

参考文献：

[1] BENDIAB G, HAMEURLAINE A, GERMANOS G, et al. Autonomous vehicles security: challenges and solutions using blockchain and artificial intelligence[J]. IEEE Transactions on Intelligent Transportation Systems, 2023, 24(4): 3614-3637.

[2] FENG Q, HE D, ZEADALLY S, et al. A survey on privacy protection in blockchain system[J]. Journal of Network and Computer Applications, 2019, 126: 45-58.

[3] HATHALIYA J J, MODI H, GUPTA R, et al. Deep learning and blockchain-based essential and parkinson tremor classification scheme[C]//Proceedings of 2022 IEEE Conference on Computer Communications Workshops (INFOCOM WKSHPS). Piscataway: IEEE Press, 2022: 1-6.

[4] LI Z Y, HE E H. Graph neural network-based bitcoin transaction tracking model[J]. IEEE Access, 2023, 11: 62109-62120.

[5] MONAMO P, MARIVATE V, TWALA B. Unsupervised learning for robust Bitcoin fraud detection[C]//Proceedings of Information Security for South Africa (ISSA). Piscataway: IEEE Press, 2017: 129-134.

[6] TOYODA K, OHTSUKI T, MATHIOPOULOS P T. Multi-class bitcoin-enabled service identification based on transaction history summarization[C]//Proceedings of IEEE International Conference on Internet of Things (iThings) and IEEE Green Computing and Communications (GreenCom) and IEEE Cyber, Physical and Social Computing (CPSCom) and IEEE Smart Data (SmartData). Piscataway: IEEE Press, 2019: 1153-1160.

[7] LIN Y J, WU P W, HSU C H, et al. An evaluation of bitcoin address classification based on transaction history summarization[C]//Proceedings of 2019 IEEE International Conference on Blockchain and Cryptocurrency (ICBC). Piscataway: IEEE Press, 2019: 302-310.

[8] BARTOLETTI M, PES B, SERUSI S. Data mining for detecting bitcoin ponzi schemes[C]//Proceedings of 2018 Crypto Valley Conference on Blockchain Technology (CVCBT). Piscataway: IEEE Press, 2018: 75-84.

[9] LI Y, CAI Y, TIAN H, et al. Identifying illicit addresses in bitcoin network[C]//International Conference on Blockchain and Trustworthy Systems. Singapore: Springer, 2020: 99-111.

[10] HU T, LIU X, CHEN T, et al. Transaction-based classification and detection approach for Ethereum smart contract[J]. Information Processing & Management, 2021, 58(2): 102462.

[11] 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.

[12] CHEN W, GUO X, CHEN Z, et al. Phishing scam detection on ethereum: towards financial security for blockchain ecosystem[C]//Proceedings of International Joint Conference on Artificial Intelligence. Piscataway: IEEE Press, 2020: 4456-4462.

[13] CHEN L, PENG J Y, LIU Y, et al. Phishing scams detection in ethereum transaction network[J]. ACM Transactions on Internet Technology, 2021, 21(1): 1-16.

[14] KE G, MENG Q, FINLEY T, et al. Lightgbm: a highly efficient gradient boosting decision tree[C]//Advances in Neural Information Processing Systems. San Francisco: Morgan Kaufmann Press, 2017: 30.

[15] YUAN Q, HUANG B Y, ZHANG J, et al. Detecting phishing scams on ethereum based on transaction records[C]//Proceedings of 2020 IEEE International Symposium on Circuits and Systems (ISCAS). Piscataway: IEEE Press, 2020: 1-5.

[16] ZHOU J, HU C, GONG S, et al. BlockGC: a joint learning framework for account identity inference on blockchain with graph contrast[J]. arXiv Preprint, arXiv: 2112.03659, 2021.

[17] LIU J L, ZHENG J T, WU J J, et al. FA-GNN: filter and augment graph neural networks for account classification in ethereum[J]. IEEE Transactions on Network Science and Engineering, 2022, 9(4): 2579-2588.

[18] KIPF T N, WELLING M. Semi-supervised classification with graph convolutional networks[J]. arXiv Preprint, arXiv:1609.02907, 2016.

[19] HAMILTON W L, YING R, LESKOVEC J. Inductive representation learning on large graphs[J]. arXiv Preprint, arXiv: 1706.02216, 2017.

[20] GUO G D, WANG H, BELL D A, et al. KNN model-based approach in classification[C]//Proceedings of OTM Confederated International Conferences. [S.l.:s.n.], 2003: 986-996.

[21] PEROZZI B, AL-RFOU R, SKIENA S. DeepWalk: online learning of social representations[C]//Proceedings of the 20th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. New York: ACM Press, 2014: 701-710.

[22] GROVER A, LESKOVEC J. Node2Vec: scalable feature learning for networks[J]. KDD: Proceedings International Conference on Knowledge Discovery & Data Mining, 2016, 2016: 855-864.

[23] RIBEIRO L F R, SAVERESE P H P, FIGUEIREDO D R. Struc2Vec: learning node representations from structural identity[C]//Proceedings of the 23rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. New York: ACM Press, 2017: 385-394.

[作者简介]

李致远（1981− ），男，河南开封人，博士，江苏大学副教授、硕士生导师，主要研究方向为区块链匿名交易追踪、物联网和软件定义网络及安全。

徐丙磊（1997− ），男，山东莱阳人，江苏大学硕士生，主要研究方向为区块链匿名可追踪、区块链地址分类。

周颖仪（2000− ），女，江苏苏州人，江苏大学硕士生，主要研究方向为区块链交易网络构建与链路预测。