

# 基于 Attention 深度随机森林的社区演化事件预测

潘剑飞<sup>1,2</sup>, 曹 燕<sup>1</sup>, 董一鸿<sup>1</sup>, 陈华辉<sup>1</sup>, 钱江波<sup>1</sup>

(1. 宁波大学信息科学与工程学院, 浙江宁波 315211; 2. 北京百度在线科技有限公司, 北京 100084)

**摘 要:** 在网络结构不断变化的同时, 社区结构也随之演化. 社区结构在不同时间片的变化可定义为四种不同的演化事件: 持续、分离、融合和消失. 本文运用网络表示学习的方法, 对网络进行图嵌入编码映射到低维向量空间中, 研究动态社区演化事件的预测. 特征方面, 在传统的社区内部属性特征、时间片间属性特性变化和前段时间片的社区演化事件的特征维度的基础上, 引入潜在结构特征表征四种演化事件, 运用随机游走和 Softmax 思想获取潜在的结构特征; 模型方面, 引入深度随机森林的策略, 同时采用 attention 机制、蒙特卡洛特征采样策略进行特征融合和特征训练, 克服了已有算法仅获取局部结构特征的缺陷. 实验在 DBLP、FACEBOOK 和 Bitcoin 数据集上, 对比 SVM、XGBOOST 和 RIDGE 模型训练, 证实了新提出的算法模型对最终预测准确率有很大的提升.

**关键词:** 社区演化; 图嵌入; 网络表示学习; 深度随机森林; attention 机制

**中图分类号:** TP311 **文献标识码:** A **文章编号:** 0372-2112 (2019)10-2050-11

**电子学报 URL:** <http://www.ejournal.org.cn> **DOI:** 10.3969/j.issn.0372-2112.2019.10.005

## The Community Evolution Event Prediction Based on Attention Deep Random Forest

PAN Jian-fei<sup>1,2</sup>, CAO Yan<sup>1</sup>, DONG Yi-hong<sup>1</sup>, CHEN Hua-hui<sup>1</sup>, QIAN Jiang-bo<sup>1</sup>

(1. Faculty of Electrical Engineering and Computer Science, Ningbo University, Ningbo, Zhejiang 315211, China;

2. Baidu Online Technology Co., Ltd., Beijing 100084, China)

**Abstract:** The internal community structure is evolving with the change of network structure. These changes in different time slices can be defined as four different evolutionary events: survive, split, fusion and disappearance. In this paper, the network representation learning method is used to map the graph embedding of the network into the low-dimensional vector space to study the prediction of dynamic community evolution events. In the features, based on the features of community internal attributes, the change of temporal attributes, and the previous community evolution events, the potential structure characteristics of the four evolutionary events are introduced and obtained by using random walk and Softmax. In the model, the strategy of deep random forest is proposed. Feature fusion and feature training are carried out by using the attention mechanism and Monte Carlo feature sampling strategy, which overcomes the shortcomings of the existing algorithms that only acquire local structural features. Finally, by comparing SVM, XGBOOST, RIDGE model training in the DBLP, FACEBOOK and Bitcoin datasets, it is confirmed that the embedding feature of community structure and the attention deep random forest model improvement have greatly improved the accuracy of final prediction.

**Key words:** community evolution; graph embedding; network representation learning; deep random forests; attention mechanism

### 1 引言

实体间的关系构成的网络存在内部结构紧密, 外部结构稀疏<sup>[1]</sup>的特点, 称为社区结构, 该结构对建立用户画像、进行商品推荐等都有有一定指引作用<sup>[2-4]</sup>. 然而,

真实世界中网络结构不断变化, 社区结构随之变化, 这种变化具有可预见性, 因此可将社区的动态变化分为持续、融合、分离和消失四类事件<sup>[5]</sup>.

传统的社区演化预测方法是通过获取社区内外的属性特征<sup>[5]</sup>建立模型, 预测下一时间片的社区事件, 往

往往仅采用社区属性及其变化等局部特性作为特征,没有考虑社区内和社区间顶点分布的整体结构特征,导致重要的社区内在特性的缺失,严重影响社区演化事件的准确性;在模型方面,传统方法往往采用单层模型<sup>[6-8]</sup>预测,预测准确性和泛化性能有待于进一步提高。

随着深度学习技术的发展,已有学者开始探索面向信息的低维稠密向量表示方法,使得网络中拓扑结构越相似的节点在该向量空间中的距离越接近.针对传统社区事件预测方法的缺陷,本文运用网络表示学习的方法,提出了基于潜在结构特征的 attention 深度随机森林模型(Attention Deep Forest based on Latent Structure,ADFLS)对社区演化事件的预测,主要贡献如下:

(1)针对社区持续、融合、分离和消失四种事件,运用网络表示学习的方法,对社区基本特征和四种事件所构成的潜在结构特征分别进行图嵌入(graph embedding)编码,结合该编码特征对社区演化事件进行预测;

(2)在预测模型上,提出了 attention 机制、特征的记忆度筛选门机制和蒙特卡洛采样策略对深度随机森林模型进行改进;

(3)真实数据集上的对比实验证实新提出的ADFLS模型对预测准确率有很大的提升。

## 2 相关工作

社区演化预测包括链接关系预测、社区的持续时间预测和整体社区事件预测三个方面。

链接关系预测.文献[9]采用决策树模型,结合社区内封闭三角形结构和开放三角形结构的特征,对社区的顶点演变推导出社区链接情况的预测.文献[10]提出社区内顶点扩展时,网络的结构将急剧改善,运用该信息能有效的提高链接关系预测的准确性。

社区的持续时间预测.文献[11]结合社区持续的过程中社区大小、社区内连接强度和社区稳定性特征的变化,运用线性回归模型预测社区的持续时间.文献[12]提出演化学习方法分析社区的行为,判定社区持续的时间和社区生命周期内的事件。

整体社区事件预测.文献[13]把社区的历史演化状态作为训练集,预测社区可能的演化状态,进而将原来的社区内顶点的变化转为社区结构变化;文献[5]从社区整体特征、点特征、点特征量统计三个维度获取特征,运用了机器学习的分类算法对时间片社区进行预测。

这三类社区演化预测事件,链路关系预测主要关注网络中成员与社区之间的各种关系;社区持续事件预测跟踪社区演化过程中的公共节点,多应用于客户

流失的可能性预测;整体社区事件预测考虑网络在结构上内聚特征,从网络结构获取局部统计特征.本文的工作属于整体社区事件预测,运用随机游走和 Softmax 思想获取潜在的结构特征,提出图嵌入的模型进行特征融合和特征训练,克服了已有算法仅获取局部结构特征的缺陷,将网络表示学习引入社区演化预测。

## 3 基本概念

表 1 为本文所用的符号与对应的描述。

表 1 符号表

符号	描述
$DG$	动态网络结构
$G_t$	$t$ 时间片的网络图
$V_t$	$t$ 时间片内网络的顶点集合
$E_t$	$t$ 时间片内网络的边集合
$C_t^i$	$t$ 时间片内第 $i$ 个社区
$V_{C_t^i}$	$t$ 时间片内 $C_t^i$ 社区结构内的顶点集合
$PR_{C_t^i}(x)$	社区 $C_t^i$ 内 $x$ 顶点在 $t$ 时间片网络结构内的 PageRank 值
$DCS(C_t^i, C_{t+1}^i)$	两个时间片 $t, t+1$ 间社区包含度
$\Delta_{C_t^i}(x)$	$t$ 时间片社区 $C_t^i$ 内顶点 $x$ 与相邻顶点构成的三角形数

**定义 1** 时间片网络:对动态网络结构  $DG$  按时间段划分,表示为  $DG = \{G_0, \dots, G_t, \dots\}$ ,其中,  $G_t = (V_t, E_t)$  为  $t$  时间片的网络结构。

**定义 2** 时间片社区:对  $G_t = (V_t, E_t), t = 1, 2, 3, \dots$  进行社区发现后得到网络的社区分布为  $\{C_t^1, C_t^2, \dots, C_t^i, \dots\}$ ,  $C_t^i$  表示在  $t$  时间片网络中的第  $i$  个社区结构。

**定义 3** 动态社区包含度  $DCS$ :两个时间片间演化社区顶点的交集的顶点数和该交集顶点的重要程度(PageRank),包含度越大,两个社区越相似.时间片  $t$  的社区  $C_t^i$  的顶点集为  $V_{C_t^i}$ ,时间片  $t+1$  的社区  $C_{t+1}^i$  的顶点集为  $V_{C_{t+1}^i}$ ,社区  $C_t^i$  与社区  $C_{t+1}^i$  的动态社区包含度为

$$DCS(C_t^i, C_{t+1}^i) = \frac{|V_{C_t^i} \cap V_{C_{t+1}^i}|}{|V_{C_t^i}|} \cdot \frac{\sum_{x \in (C_t^i \cap C_{t+1}^i)} PR_{C_t^i}(x)}{\sum_{x \in (C_t^i)} PR_{C_t^i}(x)} \quad (1)$$

其中,  $PR_{C_t^i}(x)$  表示第  $i$  个社区  $C_t^i$  内  $x$  顶点在  $t$  时间片网络结构内的重要程度。

**定义 4** 社区内核心顶点集

$$V_{C_t^i}^R = \left\{ v \mid \Delta_{C_t^i}(v) > \frac{\sum_{x \in C_t^i} \Delta_{C_t^i}(x)}{3|C_t^i(x)|}, v \in C_t^i \right\},$$

即与同一社区内所有顶点构成的三角形数量大于该社区三角形数量的平均值的顶点集合。

**定义 5** 社区内边缘顶点集

$$V_{C_i}^s = \left\{ v \mid \Delta_{C_i}(v) \leq \frac{\sum \Delta_{C_i}(x)}{3|C_i(x)|}, v \in C_i \right\},$$

即与同一社区内其他顶点构成的三角形数量小于等于该社区三角形数量的平均值的顶点集合.

$V_{C_i}^r \cap V_{C_i}^s = \emptyset, V_{C_i}^r \cup V_{C_i}^s = V_{C_i}$  将社区内核心顶点和其顶点间的关系构成的局部结构称为核心结构, 社区内边缘顶点和其顶点间的关系构成的局部结构称为边缘结构, 因此核心结构与边缘结构的顶点交集为空, 并集为该时刻社区的顶点全集.

在复杂动态网络中, 随着时间的推移, 社区可能发生持续、分离、融合和消失事件, 称之为社区演化. 不同的演化事件将导致社区的结构形态发生不同的变化.

图 1 展示了四种事件的社区演化.

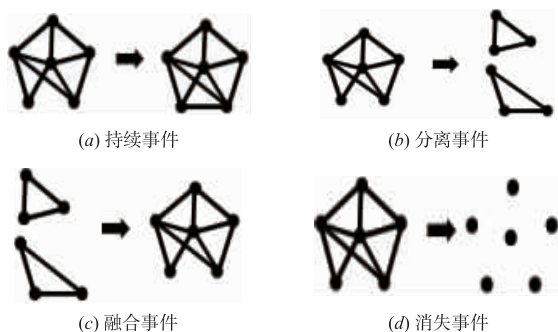


图 1 社区演化事件

## 4 特征工程

本文的主要任务是根据当前社区的属性特征和之前社区的状态变化, 预测下一时间片社区可能产生的演化事件. 特征工程是分类预测需要解决的首要问题. 提出的 ADFLS 模型中, 特征工程分为两部分: 第一部分, 运用基本特征, 从社区本身属性维度、两个相邻时间片间社区属性变化维度和前一时间片社区事件维度三个层面建模; 第二部分, 运用随机游走和 Hierarchical Softmax 思想获取潜在结构特征.

### 4.1 基本特征

社区演化事件预测任务的基本特征<sup>[7]</sup>主要体现在以下三个方面.

(1) 社区本身属性维度. 包括社区内顶点层面和社区整体层面. 顶点层面主要从重要顶点的重要程度和重要顶点间的紧密程度描述特征. 社区整体层面主要从社区内顶点的整体紧密程度获取特征.

(2) 两个时间片间社区内属性变化的维度. 即当前时刻和前一时间片同一社区之间社区本身维度特征变化.

(3) 前一时间片社区转变为当前时间片社区的事件. 主要包括社区的持续、融合、分离和消失四个基础事

件, 事件的值可用 one-hot 编码形式表示.

### 4.2 潜在结构特征

潜在结构特征包括社区消失、融合、分离和持续等四种事件所构成的特征.

**定义 6** 损失函数  $L$ . 针对不同的社区事件将核心结构或者边缘结构视为超点  $v_i$ . 从超点  $v_i$  出发, 在整个网络结构的顶点随机游走, 获取超点  $v_i$  的上下文顶点集  $\{v_{i-w}, \dots, v_{i+w}\} \setminus v_i$ , 其中  $w$  表示随机游走的步长, 根据自然语言模型建模 n-gram, 对超点  $v_i$  编码以表征上下文顶点集. 损失函数为

$$L = \text{Min}_{\varphi} \left\{ -\log \Pr(\{v_{i-w}, \dots, v_{i+w}\} \setminus v_i \mid \varphi(v_i)) \right\} \quad (2)$$

其中  $\varphi(v_i)$  表示超点  $v_i$  编码.

**定理 1** 假设每个上下文顶点  $u \in \{v_{i-w}, \dots, v_{i+w}\} \setminus v_i$  为 Huffman 树的叶子节点, 根节点为超点  $v_i$ .  $p^u$  为根节点到叶子节点的路径,  $l^u$  为根节点到叶子节点的路径的节点数,  $p^u$  路径上经过的第  $j$  个节点为  $p_j^u$ , 对应的霍夫曼编码为  $d_j^u \in \{0, 1\}$ , 其中  $j = 2, 3, \dots, l^u$ ,  $\theta_{j-1}^u$  表示非叶子节点对应的向量编码,  $\delta(x) = \frac{1}{1 + e^{-x}}$ ,  $\gamma$  为梯度学

习率,  $u \in \{v_{i-w}, \dots, v_{i+w}\} \setminus v_i$ ,  $w$  为相对超点  $v_i$  进行随机游走的窗口,  $n$  表示迭代次数, 因此超点  $v_i$  编码  $\varphi(v_i)$  为  $(\theta_{j-1}^u)^n = (\theta_{j-1}^u)^{n-1}$

$$+ \gamma [1 - d_j^u - \delta(\varphi^{n-1}(v_i)^T (\theta_{j-1}^u)^{n-1})] \varphi^{n-1}(v_i) \quad (3)$$

$$\varphi^n(v_i) = \varphi^{n-1}(v_i)$$

$$+ \gamma [1 - d_j^u - \delta(\varphi^{n-1}(v_i)^T (\theta_{j-1}^u)^n)] (\theta_{j-1}^u)^n \quad (4)$$

**证明** 对于社区的四种事件的预测, 首先获取社区内核心顶点集和社区内边缘顶点集构成的局部结构视为超点  $v_i$ , 对超点  $v_i$  的编码, 将定义 4 的损失函数进一步建模.

超点  $v_i$  编码的条件下, 上下文顶点相互独立, 得到

$$\Pr(\{v_{i-w}, \dots, v_{i+w}\} \setminus v_i \mid \varphi(v_i)) = \prod_{u \in \{v_{i-w}, \dots, v_{i+w}\} \setminus v_i} \Pr(u \mid \varphi(v_i)) \quad (5)$$

式(5)中的  $\Pr(u \mid \varphi(v_i))$  的求解时间复杂度为  $O(V)$ ,  $V$  为整个网络结构的顶点集, 每个上下文顶点为 Huffman 树的叶子节点, 根节点为  $v_i$ , 采用霍夫曼树来代替从隐藏层到输出 softmax 层的映射, 计算 softmax 概率只需要沿着树形结构进行, 最大化每一次从父节点到子节点的选择概率.  $\theta_{j-1}^u$  表示非叶子节点对应的向量编码, 基于 Hierarchical Softmax 思想找到合适的节点的词向量和所有非叶子节点  $\theta_{j-1}^u$ , 期望最大化所有样本的似然函数乘积, 将式(5)转化为式(6):

$$\Pr(u \mid \varphi(v_i)) = \prod_{j=2}^{l^u} \Pr(d_j^u \mid \varphi(v_i), \theta_{j-1}^u) \quad (6)$$

其中,

$$\Pr(d_j^u | \varphi(v_i), \theta_{j-1}^u) = \begin{cases} \delta(\varphi(v_i)^T \theta_{j-1}^u), & d_j^u = 0 \\ 1 - \delta(\varphi(v_i)^T \theta_{j-1}^u), & d_j^u = 1 \end{cases}$$

对于某一个超点  $v_i$ , 其最大似然为

$$\Pr(d_j^u | \varphi(v_i), \theta_{j-1}^u) = \delta(\varphi(v_i)^T \theta_{j-1}^u)^{1-d_j^u} [1 - \delta(\varphi(v_i)^T \theta_{j-1}^u)]^{d_j^u}$$

为了减少梯度计算量, 对每一样本使用随机梯度上升法更新梯度, 得到  $v_i$  的对数似然函数  $L$  如下

$$L = \sum_{u \in \{v_{i-1}, \dots, v_{i+1}\} \setminus v_i} \sum_{j=2}^l \left\{ (1 - d_j^u) \log[\delta(\varphi(v_i)^T \theta_{j-1}^u)] + d_j^u \log[1 - \delta(\varphi(v_i)^T \theta_{j-1}^u)] \right\} \quad (7)$$

将式(7)简化表示为

$$L(u, j) = (1 - d_j^u) \log[\delta(\varphi(v_i)^T \theta_{j-1}^u)] + d_j^u \log[1 - \delta(\varphi(v_i)^T \theta_{j-1}^u)] \quad (8)$$

$$\begin{aligned} \frac{\partial L(u, j)}{\partial \theta_{j-1}^u} &= \frac{\partial}{\partial \theta_{j-1}^u} \left\{ (1 - d_j^u) \log[\delta(\varphi(v_i)^T \theta_{j-1}^u)] + d_j^u \log[1 - \delta(\varphi(v_i)^T \theta_{j-1}^u)] \right\} \\ &= (1 - d_j^u) [1 - \delta(\varphi(v_i)^T \theta_{j-1}^u)] \varphi(v_i) \\ &\quad - d_j^u \delta(\varphi(v_i)^T \theta_{j-1}^u) \varphi(v_i) \\ &= [1 - d_j^u - \delta(\varphi(v_i)^T \theta_{j-1}^u)] \varphi(v_i) \end{aligned} \quad (9)$$

$$\frac{\partial L(u, j)}{\partial \varphi(v_i)} = [1 - d_j^u - \delta(\varphi(v_i)^T \theta_{j-1}^u)] \theta_{j-1}^u \quad (10)$$

$$\begin{aligned} \therefore (\theta_{j-1}^u)^n &= (\theta_{j-1}^u)^{n-1} \\ &\quad + \gamma [1 - d_j^u - \delta(\varphi^{n-1}(v_i)^T (\theta_{j-1}^u)^{n-1})] \varphi^{n-1}(v_i) \end{aligned} \quad (11)$$

$$\begin{aligned} \varphi^n(v_i) &= \varphi^{n-1}(v_i) \\ &\quad + \gamma [1 - d_j^u - \delta(\varphi^{n-1}(v_i)^T (\theta_{j-1}^u)^{n-1})] (\theta_{j-1}^u)^n \end{aligned} \quad (12)$$

对式(3)和式(4)运用随机梯度法进行参数优化, 收敛到最优解. 当损失函数为凸函数时, 收敛到全局最优解. 如算法 1.

#### 算法 1 潜在结构特征

输入: 图  $G$ ; 起点  $v_i$ ; 维度编码  $d$

输出: 超点  $v_i$  的编码为  $\partial \varphi(v_i)$

1.  $\varphi(v_i) = \text{RandomNum}^{1 \times d}$  /\* 随机产生  $d$  个数对超点  $v_i$  进行编码 \*/
2. While( $\Delta \varphi(v_i) > \varepsilon$ ) /\* 当超点  $v_i$  的编码稳定不产生变化是终止 \*/
3. For  $v$  in  $v_i$  do /\* 对超点  $v_i$  内每个顶点循环 \*/
4.  $\{v_{i-w}, \dots, v_{i+w}\} \setminus v_i = \text{RandomWalk}(G, v, w)$  /\* 产生超点  $v_i$  外的上下文顶点 \*/
5. BuildHuffmanTree( $v_i, \{v_{i-w}, \dots, v_{i+w}\} \setminus v_i$ ) /\* 建立超点  $v_i$  为根节点的 Huffman 树 \*/
6. For  $u \in \{v_{i-w}, \dots, v_{i+w}\} \setminus v_i$  do /\* 对每个超点  $v_i$  上下文顶点分析 \*/
7. For  $j=2:l$  do /\* 根据 Huffman 路径迭代更新, 非叶子节点的编码 \*/
8.  $g = \gamma [1 - d_j^u - \delta(\varphi(v_i)^T \theta_{j-1}^u)]$

9.  $e_i = e + g \theta_{j-1}^u$
10.  $\theta_{j-1}^u := \theta_{j-1}^u + g \varphi(v_i)$
11.  $\varphi(v_i) := \varphi(v_i) + e_i$  /\* 更新超点  $v_i$  的编码 \*/
12. Return  $\varphi(v_i)$

#### 4.3 算法收敛性

对于式(8)的损失函数分两种情况讨论.

情况 1 当  $d_j^u = 1$  时, 有

$$\begin{aligned} L(u, j) &= (1 - d_j^u) \log[\delta(\varphi(v_i)^T \theta_{j-1}^u)] \\ &\quad + d_j^u \log[1 - \delta(\varphi(v_i)^T \theta_{j-1}^u)] \\ &= \log[1 - \delta(\varphi(v_i)^T \theta_{j-1}^u)] \end{aligned} \quad (13)$$

$$\begin{aligned} \frac{\partial}{\partial \varphi(v_i)} \log[1 - \delta(\varphi(v_i)^T \theta_{j-1}^u)] &= \frac{1}{1 - \delta(\varphi(v_i)^T \theta_{j-1}^u)} (-1) \delta(\varphi(v_i)^T \theta_{j-1}^u) \\ &\quad \cdot (1 - \delta(\varphi(v_i)^T \theta_{j-1}^u)) \varphi(v_i) \\ &= -\delta(\varphi(v_i)^T \theta_{j-1}^u) \varphi(v_i) \end{aligned} \quad (14)$$

Hessian 矩阵为

$$\begin{aligned} \frac{\partial}{\partial (\varphi(v_i))^2} (\log[1 - \delta(\varphi(v_i)^T \theta_{j-1}^u)])^2 &= \frac{\partial}{\partial \varphi(v_i)} (-\delta(\varphi(v_i)^T \theta_{j-1}^u) \varphi(v_i)) \\ &= -\delta(\varphi(v_i)^T \theta_{j-1}^u) (1 - \delta(\varphi(v_i)^T \theta_{j-1}^u)) \mathbf{V} \end{aligned} \quad (15)$$

如果单个样本的特征向量为  $\varphi(v_i) = [v_{i1}, \dots, v_{in}]^T$ , 令矩阵  $\mathbf{V}$ ,

$$\mathbf{V} = \begin{bmatrix} v_{i1}^2 & \dots & v_{i1} v_{in} \\ \dots & \dots & \dots \\ v_{in} v_{i1} & \dots & v_{in}^2 \end{bmatrix} \quad (16)$$

则  $-\delta(\varphi(v_i)^T \theta_{j-1}^u) \varphi(v_i)$  的 Hessian 矩阵为  $\delta(\varphi(v_i)^T \theta_{j-1}^u) (1 - \delta(\varphi(v_i)^T \theta_{j-1}^u)) \mathbf{V}$ , 矩阵  $\mathbf{V}$  可以写成如下乘积形式:

$$\mathbf{V} = v_i v_i^T \quad (17)$$

对于任意不为 0 的向量  $v_i$  有

$$v_i^T \mathbf{V} v_i = v_i^T (v_i v_i^T) v_i = v_i^T v_i v_i^T v_i = (v_i^T v_i) (v_i^T v_i) \geq 0 \quad (18)$$

矩阵  $\mathbf{V}$  半正定.

另外由于  $\delta(\varphi(v_i)^T \theta_{j-1}^u) (1 - \delta(\varphi(v_i)^T \theta_{j-1}^u)) > 0$ , 因此 Hessian 矩阵半正定,  $\log[1 - \delta(\varphi(v_i)^T \theta_{j-1}^u)]$  为凸函数.

情况 2 当  $d_j^u = 0$  时, 有

$$\begin{aligned} L(u, j) &= (1 - d_j^u) \log[\delta(\varphi(v_i)^T \theta_{j-1}^u)] \\ &\quad + d_j^u \log[1 - \delta(\varphi(v_i)^T \theta_{j-1}^u)] \\ &= \log[\delta(\varphi(v_i)^T \theta_{j-1}^u)] \end{aligned} \quad (19)$$

Hessian 矩阵为

$$\begin{aligned} & \frac{\partial}{\partial(\varphi(v_i))^2} (\log[\delta(\varphi(v_i)^T \theta_{j-1}^u)])^2 \\ &= \frac{\partial}{\partial \varphi(v_i)} ((1 - \delta(\varphi(v_i)^T \theta_{j-1}^u)) \varphi(v_i)) \\ &= (-1) \delta(\varphi(v_i)^T \theta_{j-1}^u) (1 - \delta(\varphi(v_i)^T \theta_{j-1}^u)) \mathbf{V} \end{aligned} \quad (20)$$

矩阵  $\mathbf{V}$  半正定, 由于

$$\delta(\varphi(v_i)^T \theta_{j-1}^u) (1 - \delta(\varphi(v_i)^T \theta_{j-1}^u)) > 0 \quad (21)$$

因此,  $\log[\delta(\varphi(v_i)^T \theta_{j-1}^u)]$  是凸函数.

综上, 式(8)为凸函数, 其非负线性组合也是凸函数, 所以式(7)的损失函数为凸函数, 模型为凸优化问题, 具有全局最优解. 因此, 算法收敛.

## 5 attention 深度随机森林

### 5.1 引入 attention 机制

文献[14]提出了深度随机森林模型, 把级联后随机森林作为预测单元, 滑动窗口获取特征矩阵后, 将其输入到预测单元, 得到的预测结果作为融合后的特征, 然后结合 stacking 策略对融合后的特征进行深度学习. 然而, 该过程存在特征损失, 在 stacking 深度学习过程中未必能获取高的预测准确率. 因此引入深度学习中 attention 思想, 在特征融合后和 stacking 的每层特征获取时融入初始的特征, 增加特征记忆门, 用 stacking 的每层预测结果对特征进行选择. 根据初始层的预测赋予每个特征的重要程度即记忆度. 后续层预测过程中, 引入惩罚项  $\gamma$ , 并引用蒙特卡洛算法对特征根据当前层的特征记忆度分布进行采样. 特征记忆门结构  $\mathbf{M}$  如图 2 所示,  $\mathbf{A}$  是判断结构, 输入历史的 stacking 层的预测最大值  $R_{\max}$  和当前 stacking 层的预测值  $R$ , 当  $R > R_{\max}$  时, 进入  $\mathbf{C}$ , 输入当前的特征集  $F$ , 对当前特征集  $F$  的特征值乘以惩罚项  $\gamma > 1$ ; 当  $R < R_{\max}$  时, 进入  $\mathbf{B}$ , 输入当前的特征集  $F$  和目前最优状态的特征集  $F_{\max}$ , 用当前特征集  $F$  与  $F_{\max}$  求个差集后, 乘以惩罚项  $\gamma < 1$ , 最后进入  $\mathbf{D}$  蒙特卡洛采样结构, 采样出下一层用于预测的特征集  $F_{\text{out}}$ .

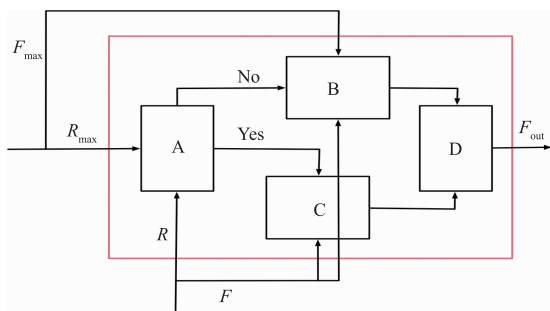


图2 记忆门结构M

基于 attention 的深度随机森林模型如图 3 所示,  $\mathbf{M}$

表示记忆门结构, 首先将原始的特征  $F_{\text{raw}}$  和特征融合后的特征  $F_{\text{mr}}$  进行合并, 然后在不断的进行 stacking 的同时, 通过 stacking 的每层预测结果结合记忆门的结构获取合适的特征, 结合历史预测情况和当前 stacking 层的预测情况, 指导下一 stacking 层的预测.

为了说明本文提出的特征记忆门和蒙特卡洛算法的机制, 引出定理 2.

**定理 2** 假设训练样本为  $D$ , 训练样本的类别为对应事件的两种类别  $C_k, k=2$ ; 特征  $a_i$  将样本集合划分为  $D_1, D_2, \dots, D_n$ , 其中  $D_i$  属于第  $C_k$  类别可表示为  $D_{ik}, \gamma$  为惩罚系数, 结合特征的信息增益和蒙特卡洛采样的策略选择特征  $f$  可表示为

$$\begin{aligned} H(D) &= - \sum_{k=1}^2 \frac{|C_k|}{|D|} \ln \frac{|C_k|}{|D|} \\ H(D|a_i) &= - \sum_{i=1}^n \frac{|D_i|}{|D|} \sum_{k=1}^2 \frac{|D_{ik}|}{|D_i|} \ln \frac{|D_{ik}|}{|D_i|} \\ & \sum_{j=1}^{f-1} \frac{\gamma * (H(D) - H(D|a_j))}{\sum_{i=1}^m \gamma * (H(D) - H(D|a_i))} \\ & \leq \text{random}(0, 1) \\ & < \sum_{j=1}^f \frac{\gamma * (H(D) - H(D|a_j))}{\sum_{i=1}^m \gamma * (H(D) - H(D|a_i))} \end{aligned} \quad (22)$$

**证明** (1) 通过对应特征的信息增益的变化判定特征重要程度. 设训练样本为  $D$ , 训练样本的类别为对应事件的两种类别  $C_k, k=1, 2$ ; 特征  $a_i$  将样本集合划分为  $D_1, D_2, \dots, D_n$ , 其中  $D_i$  属于第  $C_k$  类别可表示为  $D_{ik}$  的条件下, 数据集  $D$  的  $H(D)$  可表示为

$$H(D|a_i) = - \sum_{i=1}^n \frac{|D_i|}{|D|} \sum_{k=1}^2 \frac{|D_{ik}|}{|D_i|} \ln \frac{|D_{ik}|}{|D_i|} \quad (23)$$

因此信息增益可表示为

$$g(D, a_i) = H(D) - H(D|a_i) \quad (24)$$

获取的特征的信息增益即特征的记忆度, 并随着 stacking 的结果对特征进行不断的改进. 假设当前 stacking 层获取的特征子集为  $\{a_1, \dots, a_n\}$ , 将特征子集与当前 stacking 层预测最优的特征集的差集的特征乘以惩罚项  $\gamma, \gamma < 1$ . 因此式(24)可以表示为

$$g_\gamma(D, a_i) = \gamma * (H(D) - H(D|a_i)) \quad i \in \{1, \dots, n\} \quad (25)$$

(2) 对特征进行归一化处理. 首先根据当前层的特征的信息增益排序后的特征集为  $A = \{a_1, a_2, \dots, a_m\}$ , 结合式(25)将每个特征的信息增益归一化

$$g_\gamma(D, a_i) = \gamma * (H(D) - H(D|a_i)) \quad i \in \{1, \dots, n\} \quad (26)$$

分布在  $[0, 1]$  线段上的归一化后的特征信息增益为

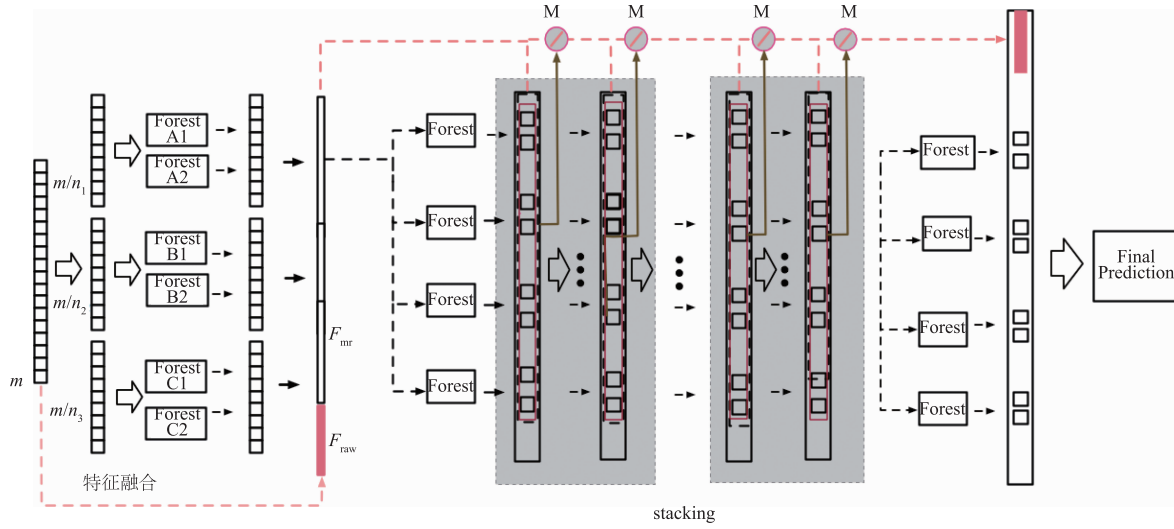


图3 attention深度随机森林

$$B = \left\{ \begin{array}{l} 0, \frac{\gamma * (H(D) - H(D | a_1))}{\sum_{i=1}^m \gamma * (H(D) - H(D | a_i))}, \\ \dots, \\ \sum_{j=1}^k \frac{\gamma * (H(D) - H(D | a_1))}{\sum_{i=1}^m \gamma * (H(D) - H(D | a_i))}, \\ \dots, 1 \end{array} \right\}$$

最终在选择特征时,结合蒙特卡洛算法进行采样,当  $\text{random}(0,1)$  表示信息增益为  $(0,1)$  之间的随机数值,且

$$\begin{aligned} & \sum_{j=1}^{f-1} \frac{\gamma * (H(D) - H(D | a_j))}{\sum_{i=1}^m \gamma * (H(D) - H(D | a_i))} \\ & \leq \text{random}(0,1) \\ & < \sum_{j=1}^f \frac{\gamma * (H(D) - H(D | a_j))}{\sum_{i=1}^m \gamma * (H(D) - H(D | a_i))} \end{aligned}$$

时选择特征  $f$ .

具体算法如算法 2.

#### 算法 2 Stacking 预测算法

输入:特征融合阶段和初始阶段融合特征集  $\text{train\_datas}$ ;  $\text{train\_labels}$ ;  $\text{test\_datas}$

输出:stacking 层的预测结果

1.  $\text{fea\_imp} = \text{classifier. fit\_transform}(\text{train\_datas}, \text{train\_labels}, \text{test\_datas})$   
/\* 通过级联的随机森林获取输入的特征的重要程度 \*/
2. **While**(1) { /\* 进入特征筛选循环 \*/
3.  $\text{fea\_ins} = \text{train\_fea. subtraction}(\text{maxtrain\_fea})$   
/\* stacking 层特征集与最优解特征集差集 \*/
4. **if** ( $\text{result}[0] > \text{maxresult}$ ) {  $\text{fea\_imp}[\text{train\_fea}] = \text{fea\_imp}[\text{train\_}$

- $\text{fea}] * \gamma$   
/\* 前一 stacking 层的预测结果大于目前最优解时,对应特征记忆度增加  $\gamma$  倍 \*/
5. **else** {  $\text{fea\_imp}[\text{fea\_ins}] = \text{fea\_imp}[\text{fea\_ins}]$  }  
/\* 否则,差集特征记忆度减少  $\gamma$  倍 \*/
6. **for** ( $\text{fea}$  in  $\text{fea\_imp}$ ) {  $\text{end} = \text{start} + \text{fea}/\text{sum}(\text{fea\_imp})$ ;  $\text{star} = \text{end}$ ;  $\text{fea\_sort} = [\text{start}, \text{end}]$  }  
/\* 特征归一化到  $[0,1]$  的范围内 \*/
7.  $\text{num} = \text{random. uniform}(0,1)$
8. **for** ( $\text{fea}$  in  $\text{fea\_sort}$ ) { **if** ( $\text{fea\_sort}[0] < \text{num} \leq \text{fea\_sort}[1]$ );  $\text{train\_fea. add}(\text{fea})$  }  
/\* 根据特征归一化的区域进行蒙特卡洛采样 \*/
9.  $\text{result} = \text{classifier. fit\_transform}(\text{train\_datas}, \text{train\_labels}, \text{test\_datas})$   
/\* 用采样到的特征进行预测得到当前 stacking 层的结果  $\text{result}$  \*/
10. **if** ( $\text{result}[0] \geq \text{maxresult}$ ) {  $\text{maxresult} = \text{result}[0]$ ;  $\text{maxtrain\_fea} = \text{train\_fea}$  }  
/\* 得到当前预测的最大集,更新最优的特征集 \*/
11. **if** ( $\text{result}[0] = \text{maxresult}$  and  $\text{result}[0] = \text{pre\_result}$ ) {  $\text{num\_max} = \text{nummax} + 1$  }  
/\* 当预测结果持续达目前最大值,循环结束 \*/
12. **if** ( $\text{num\_max} > 5$ ) { **break** } }
13. **return**  $\text{result}[1]$  /\* 返回测试集结果 \*/

#### 5.2 时间复杂性分析

本文工作主要分为特征工程与 attention 深度随机森林两部分. 特征工程中 n-gram 算法更新超点,使用 Softmax 方法训练,将每个超点分配到二分类树的叶子节点上,使用 Huffman 对节点编码,将频繁出现的节点的路径设置较短. 式(6)中  $\text{Pr}(u | \varphi(v_i))$  的时间复杂度从  $O(V)$  降到了  $O(\log |V|)$ , 其中  $|V|$  为超点的数量. 因此,特征工程的时间开销为  $O(TV \log |V|)$ ,  $T$  为迭代次数; attention 深度随机森林部分的时间复杂度为  $O(TV \log |V|)$ ,  $m$  为特征维度. 总时间复杂度为  $T(n) =$

$$O(TV \log |V|) + O(m) + O(\log m).$$

## 6 实验

### 6.1 数据集和对比模型

为了验证算法的适用性,实验使用的数据包括文章索引的 DBLP 数据、社交常用的 FACEBOOK 数据和金融交互的信任网络比特币交易数据<sup>[15]</sup>.

(1) DBLP 的数据作者之间的引用数据,以年限进行划分,将数据集划分为 10 份时间片数据.

(2) FACEBOOK 数据,通过广度搜索策略从 FACEBOOK 的网络中爬出关系数据,根据建立关系的时间顺序,然后将数据集划分为 10 份时间片的数据.

(3) Bitcoin 数据是比特币交易的信任网络,为防止欺诈和高风险交易,根据进行交易的时间顺序,将数据集划分为 10 份时间片的数据.

为了评估提出的 ADFLS 模型,选择了 SVM<sup>[6]</sup>、XGBOOST<sup>[7]</sup>、RIDGE<sup>[8]</sup> 和 DEEPFOREST<sup>[4]</sup> 四个模型进行对比实验.

### 6.2 实验模型参数

初始情况特征的维度为  $m$  维,输入模型首先经过  $m/n_1$ 、 $m/n_2$  和  $m/n_3$  的窗口滑动后得到的特征维度分别为  $(m - m/n_1 + 1) * m/n_1$ 、 $(m - m/n_2 + 1) * m/n_2$  和

$(m - m/n_3 + 1) * m/n_3$  的特征矩阵;然后将其输入随机森林分别进行特征融合得到  $(m - m/n_1 + 1) * nclass$ 、 $(m - m/n_2 + 1) * nclass$  和  $(m - m/n_3 + 1) * nclass$  的特征矩阵,最后将三个特征矩阵转为一维矩阵形式,完成整个前期过程的特征融合阶段,其中  $n_1$ 、 $n_2$  和  $n_3$  控制窗口的大小,同时调整滑动步长来增加选择特征的随机性, $nclass$  表示目标划分类数.

本小节主要对 ADFLS 模型中所涉及到的主要超参数在准确率上造成的影响进行分析.

#### 6.2.1 编码长度

考察潜在结构编码长度对事件预测的准确性的影响,窗口大小  $n_1 = 2, n_1 = 3, n_1 = 4$ ,窗口滑动步长为 2.在对超参数进行实验时,其他参数保持默认不变.

图 4 描述了 DBLP 数据集下潜在结构特征编码长度与 ADFLS 框架预测的准确性的分布情况.针对消失、融合、分离和持续四种不同的事件,编码长度为 35、10、10 和 40 时,达到预测的最优解分别为 0.7226、0.7194、0.9367 和 0.7407.可见,当特征编码长度过长时,在训练过程中容易过拟合;当结构特征编码长度较短时,该编码未能准确的描述出结构内外顶点的分布情况.因此,潜在结构编码长度对事件预测的准确率有一定的影响,但是编码长度在一定范围内,事件的预测准确率存在优解.

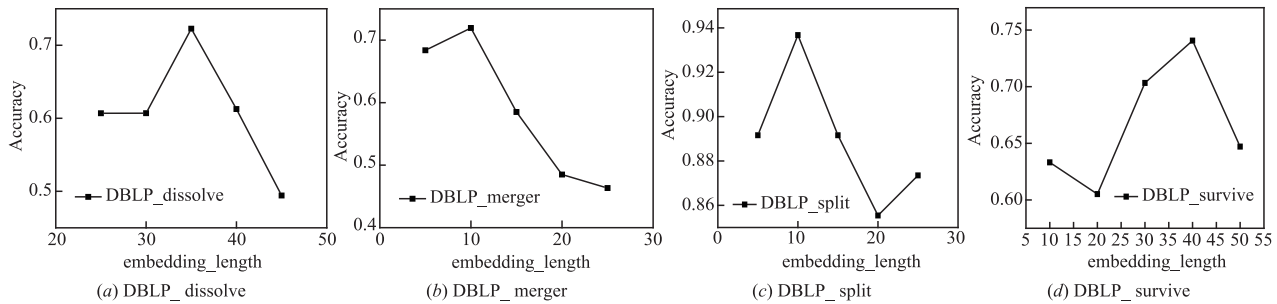


图4 DBLP 潜在结构编码长度与预测准确率分布

#### 6.2.2 滑动窗口大小及步长

在深度模型的特征融合的过程中,通过滑动窗口单步滑动和随机森林交叉验证的特征融合策略,效率比较低,因此考虑控制滑动的步长进行特征采样.通过实验的方法定量分析滑动窗口大小及步长对实验结果

准确率放入影响,使用 Bitcoin 数据集分别对社区演化的四个事件进行实验,社区演化事件参数选择如表 2 所示,当 stacking 预测的结果提升时,惩罚项  $\gamma = 10$ ;当 stacking 预测的结果下降时,惩罚项  $\gamma = 1/10$ .

表 2 社区演化事件参数及窗口滑动步长选择表

$n_1$	$n_2$	$n_3$	融合事件				消失事件				分离事件				持续事件			
			滑动窗口步长															
			2	3	4	5	2	3	4	5	2	3	4	5	2	3	4	5
2	3	4	61.5	61.5	61.5	55.1	61.2	70.2	61.1	61.2	66.6	65.5	67.7	64.5	66.6	64.5	65.9	65.2
3	4	5	61.5	57.8	59.0	55.1	65.7	73.7	67.7	65.7	66.6	66.6	68.7	66.6	65.2	65.9	65.9	65.2
4	5	6	61.5	62.8	59.0	60.3	66.2	66.7	71.7	67.2	68.7	66.6	66.6	66.6	63.1	65.2	65.2	66.7
5	6	7	59.0	55.1	62.8	57.7	66.2	67.2	69.7	67.2	66.6	66.6	67.7	67.7	65.9	65.9	65.2	65.9
6	7	8	57.7	59.0	59.0	61.5	69.2	63.6	68.2	68.7	65.5	69.8	70.9	67.7	68.6	64.5	64.5	65.9

表 2 显示社区融合事件中,窗口滑动步长为 5 时预测准确率较差;消失事件中窗口滑动步长为 3 时准确率较好;分离事件预测的准确率最低为 64.5,最高为 70.9,变化不大;社区持续事件的预测准确率没有显著变化。

总体来说,attention 深度随机森林模型的特征融合过程中,窗口的大小及窗口滑动步长的变化虽然对社区演化事件的预测准确率有一定的影响,但影响较弱。

### 6.3 特征的重要程度分布

运用 attention 深度随机森林模型对三种数据集进行测试.图 5 显示了当前状态下,特征的重要程度分布,表 3 为图 5 中各符号的含义。

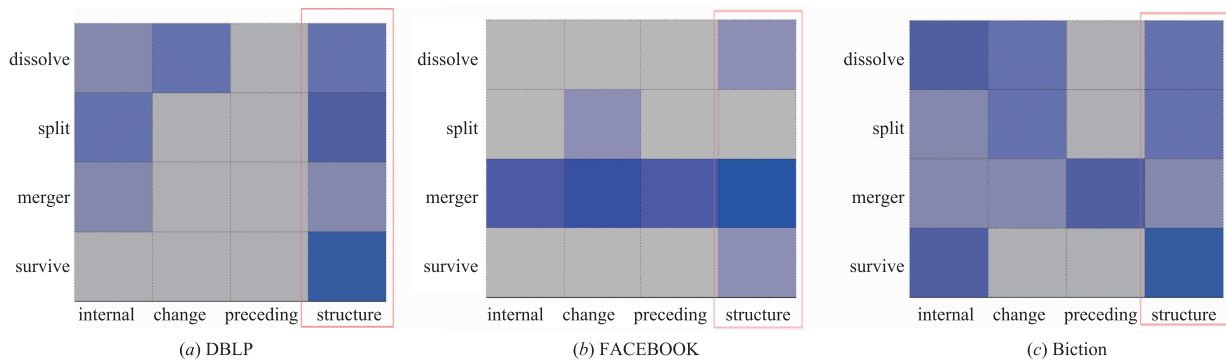


图5 特征重要程度分布

在图 5 中颜色越深表示该类型的特征重要程度越高.从三个数据集的特征重要程度可知,潜在结构特征(横坐标中的 structure,红框部分)大部分区域颜色较其他特征更深,说明该特征对预测事件具有重要意义。

表 3 图 5 中各符号的含义

横坐标		纵坐标	
符号	含义	符号	含义
internal	社区本身属性特征	survive	持续事件
change	两时间片间社区属性变化特征	merger	融合事件
preceding	两时间片间演化的事件特征	split	分离事件
structure	潜在结构特征	dissolve	消失事件

### 6.4 特征预测结果

通过 SLPA 标签传递算法对 DBLP、FACEBOOK 和 Bitcoin 的时间片数据分别进行社区发现,对两个时间片的社区运用相似度比较策略如式(2)所示,获取两个时间片对应的社区后使用基本特征后分别用 SVM、XGBOOST 和 RIDGE 三个模型对社区下一时间片的社区事件的预测。

图 6~8 分别描述了在 DBLP、FACEBOOK 和 Bitcoin 三个数据集上对消失、分离、融合和持续四种社区变化事件的预测结果。

图 6~8 横坐标中各符号的含义如下。

internal:社区本身属性特征;

+ change:社区本身属性和两时间片间社区属性变化特征;

+ preceding:社区本身属性、两时间片间社区属性变化和两时间片演化的事件特征;

+ structure:社区本身属性、两时间片间社区属性变化、两时间片演化的事件和潜在结构特征。

从前三个特征维度和非深度模型 SVM、XGBOOST 和 RIDGE 对四种社区事件的预测结果进行分析,+ change/internal 与 + preceding/+ change 特征维度比较可得 change 特征和 preceding 特征在不同数据集的不

同事件下对准确率影响的程度不同,change 特征的重要程度大于 preceding 特征的重要程度.新模型 ADFLS 的预测结果整体上优于文献[7]的深度模型,说明引入 attention 机制的新模型预测能力高于深度随机森林的模型策略;最后融入了 structure 的特征,整体的预测准确率有大幅度的提升,进一步说明了新提出的潜在结构特征对社区事件预测有很大的指引作用。

## 7 结束语

为提高社区演化事件的预测的准确率,提出了基于潜在结构特征的 attention 深度随机森林模型(ADFLS).特征方面,在基本特征的基础上,增加了表征社区持续、融合、分离和消失四种事件的潜在结构特征,根据随机游走的思想获取局部周围的顶点分布,运用 Hierarchical Softmax 的思想对潜在结构进行编码;模型方面,针对深度随机森林在特征融合过程中的特征损失和 stacking 过程不能获得高预测准确率的问题,将原始特征和融合后的特征进行合并后,用 attention 和记忆门机制防止训练过拟合的同时,采用蒙特卡洛和自学习策略获取预测的较高的预测准确率.本文针对的是动态社区的背景下,提出的 attention 深度随机森林的模型适用于解决社区演化事件的预测,未来研究可脱离

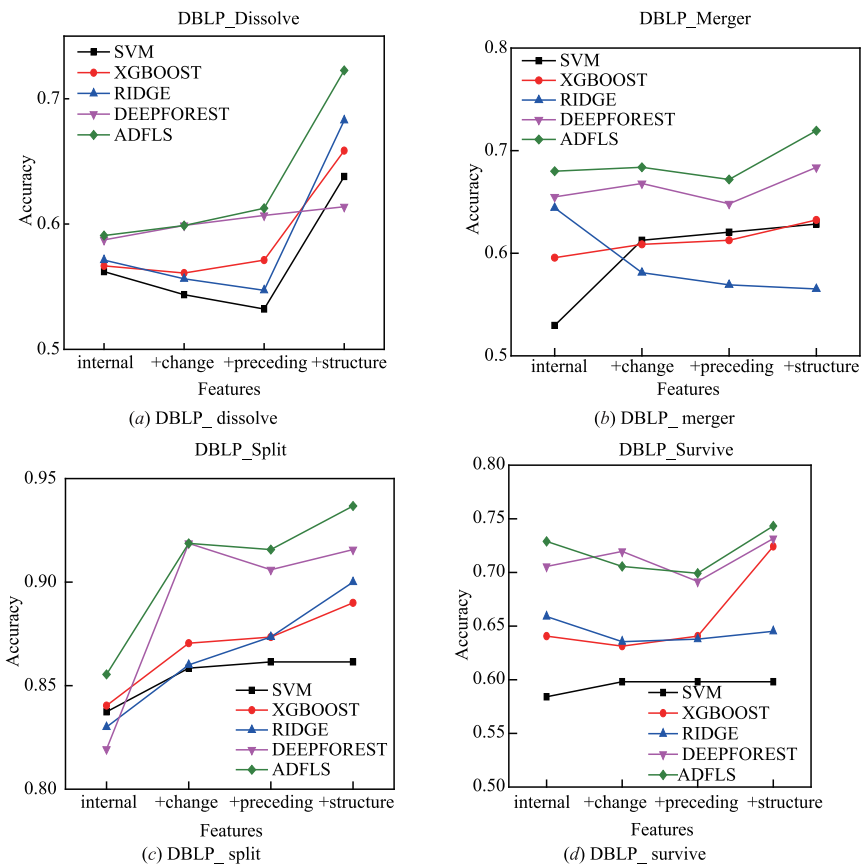


图6 DBLP数据集预测准确率分析

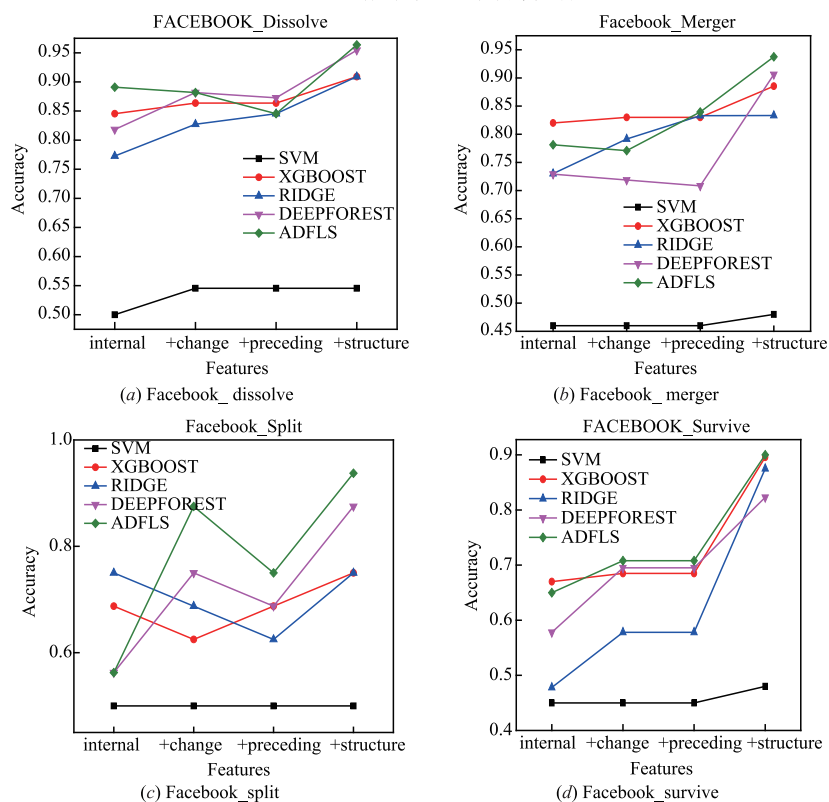


图7 FACEBOOK数据集预测准确率分布

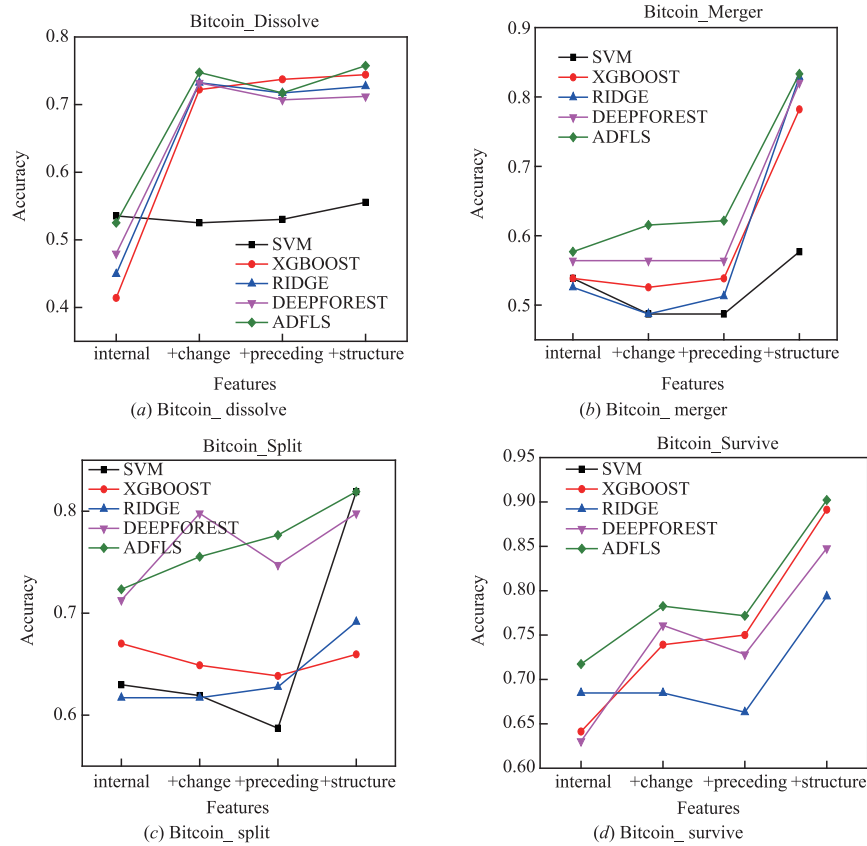


图8 Biction 数据集预测准确率分布

局限的背景,形成通用的模型框架,并将 stacking 的逐层预测赋予时间序列的属性,用于解决有前后关系的应用问题.

#### 参考文献

- [1] YUANZHUO W, XIAOLONG J, XUEQI C. Network big data: Present and future[J]. Chinese Journal of Computers, 2013, 36(6): 1125 - 1138.
- [2] XIE J, SZYMANSKI B K, LIU X. Slpa: Uncovering overlapping communities in social networks via a speaker-listener interaction dynamic process [A]. IEEE 11th International Conference on Data Mining Workshops [C]. US: IEEE, 2011. 344 - 349.
- [3] 马慧芳, 谢蒙, 何延年, 等. 基于核心标签的可重叠微博网络社区划分方法 [J]. 电子学报, 2017, 45(4): 769 - 776.  
MA Hui-fang, Xie Meng, HE Yan-nian, et al. An overlapping microblog community detection algorithm via core tags [J]. Acta Electronica Sinica, 2017, 45(4): 769 - 776. (in Chinese)
- [4] 王诗懿, 董一鸿, 李志超, 等. 大规模复杂网络下重叠社区的识别 [J]. 电子学报, 2015, 43(8): 1575 - 1582.  
WANG Shi-yi, DONG Yi-hong, LI Zhi-chao, et al. The identification of overlapping communities in large-scale complex networks [J]. Acta Electronica Sinica, 2015, 43(8): 1575 - 1582. (in Chinese)
- [5] BRODKA P, SAGANOWSKI S, KAZIENKO P. GED: the method for group evolution discovery in social networks [J]. Social Network Analysis and Mining, 2013, 3(1): 1 - 14.
- [6] WU J, YANG H. Linear regression-based efficient SVM learning for large-scale classification [J]. IEEE Transactions on Neural Networks and Learning Systems, 2015, 26(10): 2357 - 2369.
- [7] CHEN T, HE T, BENESTY M, et al. Xgboost: Extreme Gradient Boosting [OL]. <https://github.com/dmlc/xgboost>. Accessed: 2017-10-18.
- [8] MCDONALD G C. Ridge regression [J]. Wiley Interdisciplinary Reviews: Computational Statistics, 2009, 1(1): 93 - 100.
- [9] BACKSTROM L, HUTTENLOCHER D, KLEINBERG J, et al. Group formation in large social networks: membership, growth, and evolution [A]. Proceedings of the 12th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining [C]. US: ACM, 2006. 44 - 54.
- [10] FENG X, ZHAO J C, XU K. Link prediction in complex networks: A clustering perspective [J]. The European

Physical Journal B, 2012, 85(1):3.

- [11] GOLDBERG M, MAGDON-ISMAIL M, NAMBI RAJAN S, et al. Tracking and predicting evolution of social communities [A]. IEEE Third International Conference on Privacy, Security, Risk and Trust and IEEE Third International Conference on Social Computing [C]. US: IEEE, 2011. 780 – 783.
- [12] AU W H, CHAN K C C, YAO X. A novel evolutionary data mining algorithm with applications to churn prediction [J]. IEEE Transactions on Evolutionary Computation, 2003, 7(6): 532 – 545.
- [13] BRODKA P, KAZIENKO P, KOLOSZCZYK B. Predicting group evolution in the social network [A]. International Conference on Social Informatics [C]. Berlin, Heidelberg: Springer, 2012. 54 – 67.
- [14] ZHOU Z H, FENG J. Deep forest: Towards an alternative to deep neural networks [J]. Arxiv Preprint, 2017, arXiv: 1702.08835.
- [15] ANDERSEN R, CHUNG F, LANG K. Local graph partitioning using pagerank vectors [A]. The 47th Annual IEEE Symposium on Foundations of Computer Science (FOCS'06) [C]. US: IEEE, 2006. 475 – 486.

#### 作者简介



潘剑飞 男, 1991 年生. CCF 学生会员, 2018 年获宁波大学计算机技术专业硕士学位, 现为百度公司算法工程师, 主要研究方向为大数据、数据挖掘.



曹燕 女, 1993 年生. CCF 学生会员, 宁波大学信息科学与工程学院硕士, 主要研究方向为大数据、数据挖掘.



董一鸿 (通信作者) 男, 1969 年出生于浙江宁波. 博士, CCF 会员, 宁波大学教授, 主要研究方向为大数据处理、数据挖掘和人工智能等.  
E-mail: dongyihong@ nbu. edu. cn



陈华辉 男, 1964 年出生于浙江鄞州. 博士, CCF 会员, 宁波大学教授, 主要研究方向为数据库技术、流数据处理等.  
E-mail: chenhuahui@ nbu. edu. cn



钱江波 男, 1974 年出生于浙江宁波. 博士, CCF 高级会员, 宁波大学教授, 主要研究方向为数据库技术、流数据处理、多维数据索引技术等.  
E-mail: qianjiangbo@ nbu. edu. cn