基于编辑行为码的图概要技术

王 雄[1],董一鸿[1],潘剑飞[2],陈华辉[1],钱江波[1]
(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: 图数据的处理面临庞大规模和复杂结构的制约. 图的概要化,旨在寻找一组简洁的超图或稀疏图,阐明原始图的主要结构信息或变化趋势. 针对属性图提出了基于编辑行为码的概要模型,遵循最小描述长度原理(Minimum Description Length,MDL),将结构的相似性和属性的相似性统一为存储代价,构建编辑行为码. 在此模型基础上提出了 Greedy 算法和 Random 算法,存储属性和结构的编辑信息,生成高质量的超图,并支持原始图的重构. 实验结果表明本文提出的概要模型和算法相比于其他图概要算法,在压缩率和时间代价等指标上具有一定的优越性.

关键词: 图概要; 编辑行为码; 超图; 可视化
中图分类号: TP391 **文献标识码:** A **文章编号:** 0372-2112 (2020)12-2434-10
电子学报 URL: http://www.ejournal.org.cn **DOI:** 10.3969/j.issn.0372-2112.2020.12.020

Graph Summarization Technique Based on Edit Behavior Coding

WANG Xiong[1], DONG Yi-hong[1], PAN Jian-fei[2], CHEN Hua-hui[1], QIAN Jiang-bo[1]
(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 processing of graph data is restricted by large scale and complex structure. The graph summarization is to find a set of simple hyper-graphs or sparse graphs that clarify the main structural information or change trend of the original graph. According to the principle of minimum description length(MDL), a summary model based on edit coding is proposed for attribute graphs, which unifies the similarity of structure and attribute as storage cost to construct editing behavior code. Based on this model, a greedy algorithm and a random algorithm are proposed, which store the editing information of attribute and structure, generate high-quality hyper-graphs, and support the reconstruction of original graphs. The experimental results show that the proposed model and algorithms in this paper have some advantages over other summarization algorithms in terms of compression rate and time cost.

Key words: graph summarization; edit coding; hyper-graph; visualization

1 引言

现实世界中存在以图结构表达的数据,如道路网络、社会网络、生物网络等,其中顶点表示实体,边表示实体之间的联系. 大规模的图结构数据具有上亿顶点和千亿条边,例如 Facebook 的社交网络超过 30 亿用户,每天邮件投递数超过千亿,所产生的图数据无法直接加载到内存处理,大多数图分析算法无法直接作用于原始图提炼潜在的结构特征,给社会网络分析和数据挖掘带来了挑战. 作为解决方案之一,针对大规模图数据的概要化技术应运而生. 图的概要化[1,2](graph summarization)运用各种图形操作技术,寻找一组简洁的超图或稀疏图[3],阐明原始图的主要结构信息或变化趋势,代替原始大图进行数据分析.

结构图根据拓扑相似性进行概要化,即遵循公共邻居顶点越多顶点越相似的规则进行分组,典型的算法有采样聚类[4]、多路切割[5]、谱方法[6]、随机游走策略[7]等. 属性图考虑属性和结构的一致性,常见方案是在某一维度上计算顶点相似性,将之转化为另一维度上进行衡量[8]. 这种方案并不能解决属性和结构相矛盾的问题,无法确定两者之间的权重以及转化逻辑的合理性.

高质量的属性图概要满足:同组内的顶点密集连接,不同组之间的顶点稀疏;同组内的顶点属性尽量相

收稿日期:2019-03-06;修回日期:2020-06-15;责任编辑:孙瑶
基金项目:浙江省自然科学基金(No. LY20F020009, No. LZ20F020001);国家自然科学基金(No. 61502133)

似,不同组内的属性尽量差异化;现有的图概要技术需预先设定超级顶点或者分组的数量,当用户对数据没有充分了解时,则无法生成高质量的概要图,而且均未考虑概要图的重构. 针对以上问题,本文遵循 MDL 原理,结合加权编辑距离(Weight Levenshtein Distance, WLD)和 Lempel-Ziv(LZW)编码思想,提出一种基于编辑行为码的属性图概要模型(Attribute graph Summarization based on Edit Behavior Coding, ASEC),并基于不同的应用场景提出 Greedy 和 Random 两种概要算法:Greedy 算法旨在寻找高压缩率的概要图;Random 算法偏好于快速实时概要化.

(1)引入新的距离函数,提出编辑行为码(Edit Behavior Coding, EBC)将结构和属性的一致性转化为编辑行为代价,基于无监督方法指导顶点分组,降低用户与数据的交互性,并支持复杂属性图的概要化.

(2)提出概要图重构的解码方案,帮助用户更好的分析理解原始图,据我们所知,这是第一个针对属性图还原的概要技术.

2　相关工作

近年来许多学者对图概要技术进行深入研究,在不同的应用领域和背景下提出各种概要技术,用于分析理解原始图. 针对简单图数据,根据图的拓扑结构相似性进行概要化,例如 Navlaka[9] 第一个基于 MDL 信息论在有界误差内生成概要图,SlashBurn[10] 寻找高度顶点和辐条,对顶点进行排序,递归地将图分割成集线器和辐条,以 MDL 的原则生成概要模型. 针对属性图,k-SNAP[11] 基于属性的划分寻找最大兼容关系属性图,CANAL[12] 在 k-SNAP 的基础上,针对顶点属性为连续值时,利用链接结构和隐藏的信息实现自动化分类,评估概要图的兴趣度,帮助用户自动找出有意义的概要图. AGSUM[13] 采用压缩和概要技术,通过编码寻找概要图的最小描述.

上述方法只从结构或属性一个维度进行概要化,另外一些同时兼顾俩者之间的相似性,S-Entropy[14] 通过图形增强将属性维度转为属性顶点,新增属性边,提出一种信息熵度量顶点的相似性. SAC[15] 是一种基于图的拓扑结构和顶点属性相似度的聚簇算法,以增广图上的随机游走距离作为度量进行聚簇. 为避免结构和属性相似性度量的人为设计,Zneika[16] 将 RDF(Resource Description Framework)转化为多类型图,并将实体解析问题化为多类型图概要. 文献[17]针对动态图流提出一种哈希技术的图流草图. Qu 等人[18] 基于扩散半径和范围的概念,定义动态网络的兴趣度量,捕获随时间变化的关键顶点或边来概要化动态网络.

目前的概要技术要求属性集能以 one-hot 编码表示. 实际应用中属性结构复杂,不同属性的距离刻度不同,如合作者网络中的顶点具备年龄、身高等连续数值,而职称则具有递进关系,无法基于 0/1 编码表示. 部分图概要技术需要用户手动进行顶点分组设置才能生成概要图,间接提高了概要化技术使用标准.

针对以上问题,本文基于属性图存储的代价,设计一种新的属性度量方法——加权编辑距离,并将其结构和属性的差异化转化为编辑行为码,以编码-字典的形式计算存储代价,作为损失函数指导顶点的分组,无需预设分组数量,并保留了属性图的结构和属性特征,更易获得高质量概要图.

3　问题定义

属性图在现实世界中普遍存在,如社会网络中的用户兴趣,基因交互网络中的表达信息,交通网络的车流量、限速等. 事实上,实际案例往往要求综合考虑结构和属性的相似性,将结点划分到不同的分组.

在属性图数据中,顶点代表实体,边表示实体间的关系,实体往往具有各种维度的属性,在属性图中以不同的形式所表现,下面给出属性图的定义.

定义 1　属性图 G 是一个四元组 $G=(V,E,A,F)$,其中 $V=\{v_1,v_2,v_3,\cdots,v_n\}$ 是所有顶点的集合,$E=\{(v_i,v_j)\mid 1\leqslant i,j\leqslant m,i\neq j\}$ 表示所有边的集合,A 表示所有属性的集合,对于顶点 $v_i\in V$ 的属性为 $[a_1(v_i),a_2(v_i),a_3(v_i),\cdots,a_k(v_i)]$;函数 $F=(f_1,f_2,\cdots,f_k)$ 是 k 个函数的集合,f_k 对于每个顶点 $v_i\in V$ 的属性 a_k 赋予了具体的属性值和形式.

定义 2　对于属性图 G,图的顶点分组 $P=\{G_1,G_2,G_3,\cdots,G_p\}$ 当且仅当满足以下条件:

① $\forall G_i\in G,i\leqslant p,V(G_i)\in V,V(G_i)\neq\varnothing$;

② $U_{i=1}^{p}V(G_i)=V(G)$;

③ $\forall G_i,G_j\in G,i\neq j,V(G_i)\cap V(G_j)=\varnothing$.

P 中的每个元素 G_i 称作属性图 G 的一个分组,也可以称为超点,内部聚集了结构和属性相似的顶点集合,$|P|$ 表示分组的大小或超点的个数.

定义 3　给定属性图 $G=(V,E,A,F)$,概要图被定义为按照 $P=\{G_1,G_2,G_3,\cdots,G_p\}$ 进行映射,形成的子图 $G_s=(V_s,E_s)$ 和编辑行为码字典 D,其满足条件:

① $\forall v_m\in G_p,U_{j=1}^{m}v=V_s$;

② $E_s=\{(G_i,G_j)\mid 1\leqslant i,j\leqslant p,i\neq j\}$.

原始图中的每个顶点都被分配到唯一的分组中,同一分组中所有顶点构成一个超点,超点之间的边缘简称为超边,对于超点内部的任意两个顶点均存在边. 图概要,就是制定顶点之间相似性度量函数,选择迭代策略进行顶点的合并,生成概要图 G_s 代替原始图进行应用分析.

4 概要方法

本文主要研究属性图的概要化,利用最小描述长度原理,结合 WLD 距离和 LZW 编码思想,将结构和属性的一致性转化为存储代价,并提出两种基于编码代价的属性图概要算法.

4.1 MDL 原理

MDL 原理由 Risanen 于 1978 年研究通用编码时提出,基本原理是对一组给定的数据采用某种模型进行编码压缩,保存编码和模型,数据总描述长度等于编码长度加上模型所需的数据长度.

本文提出 ASEC 模型,通过编码的存储代价度量顶点之间的相似性,保存概要图重构的重要信息.

对于属性图 $G\rightarrow G_s=(V_s,E_s)$ 的概要化过程中,按照 WLD 距离生成的一组二进制信息编码,实现对边和顶点属性两类对象的增加、删除、替换、清除、插入和交换六种编辑行为(Edit Behavior,EB). 编辑行为码主要有两部分组成:编辑行为和编辑对象,编辑对象中分为顶点对象和属性对象,针对顶点对象的编辑行为有“+”,“-”,代表了边的增加和删除;针对属性对象的编辑行为有四种,表示替换,清除,插入,交换,总计六种编辑行为,每一次编辑行为和对象进行二进制编码表示.

4.2 结构概要

属性图的概要化需考虑结构和属性的一致性. 顶点的结构相似性主要体现在邻居顶点,即顶点满足共同邻居越多,其结构相似性就越大,合并相似顶点生成超点,减少冗余边. 当顶点的属性和邻居顶点完全一致,顶点在图数据中的作用和特征完全等价. 设 u,v 分别是超点,$|\Pi_{u,v}|$ 表示超点 (u,v) 的所有可能组合的边数量,$|A_{u,v}|$ 表示超点 (u,v) 之间实际存在的边数.

定理 1 给定简单图 G,合并顶点 (u,v),其编码代价最小为 $\min\{(|\Pi_{u,v}|-|A_{u,v}|)+1,|A_{u,v}|\}$.

证明 设存储边的编码代价为 1,合并顶点 u,v 时,采取两种编辑行为进行编码,当添加超边时需在编辑行为码中以‘-’删除边,‘-’的边数为不存在的边数,即 $|\Pi_{u,v}|-|A_{u,v}|$,其编码代价合为 $|\Pi_{u,v}|-|A_{u,v}|+1$,当删除超边时则需在编辑行为码中以‘+’新增边,边的数量为 (u,v) 之间真实存在的边,代价和为 $|A_{u,v}|$,因此合并顶点 (u,v),其存储代价最小为 $\min\{(|\Pi_{u,v}|-|A_{u,v}|)+1,|A_{u,v}|\}$. 证毕

基于定理 1,当选择第一种编辑方式时,需新增超边,将超边代价放入 Gs 中进行计算,推出其结构的编辑行为码最小数 $\min\{(|\Pi_{u,v}|-|A_{u,v}|),|A_{u,v}|\}$,给出合并顶点 (u,v) 的结构编码代价度量函数:

$$\begin{aligned}\mathrm{Cost}(u,v)_{结构}=\min\{(|\Pi_{u,v}|-|A_{u,v}|),|A_{u,v}|\}\\ \times L(D)+\mathrm{Cost}(Gs(u,v))\end{aligned}\tag{1}$$

其中 $L(D)$ 表示一次编辑行为码的码长,$Gs(u,v)$ 表示在概要图 Gs 中 (u,v) 合并后的超点是否新增超边,当采用“+”的编辑行为时 $Gs(u,v)=0$,否则为 1.

如图 1 所示,在进行结构概要化时,(f,g) 顶点具有相同邻居顶点直接合并,在针对顶点 (a,h) 合并时,存在两种编辑行为描述这一过程,例如用“-,(b,h)”或者“+,(a,b)”表示其编辑行为,选择代价和最小的“+,(a,b)”编辑行为,在合并顶点 (c,d,e) 时,用“-,(b,e)”或者“+,(c,b)”,“+,(d,b)”来描述这一过程,两者代价相等,为了概要图的紧凑性,选择其“-,(b,e)”生成编辑行为码. 前三位编辑行为码分别表示编辑操作‘-’,‘+’. 后六位表示编辑的顶点对象,按照 a 到 h 字母序进行二进制编码.

最终,顶点的结构压缩生成概要子图 Gs 和一组编辑行为码,观察图 1 发现,超点 (D,B) 具有相同的邻居顶点,再次合并.

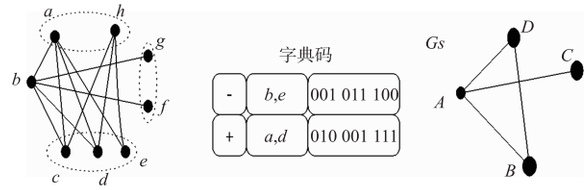

图1 结构概要化和编码

4.3 属性概要

4.3.1 WLD 距离

属性概要化时,某一顶点的所有属性的集合,以统一的顺序表示称为该顶点的属性序列,集合中元素个数称为属性的长度. 本文提出加权编辑距离 WLD,将顶点属性序列 α 变换为顶点属性序列 β 所需的清除、插入、替换,交换等编辑操作的加权距离. 使用 WLD 衡量顶点属性之间的相似度,见式(2):

$$\begin{aligned}&\mathrm{WLD}_{\alpha,\beta}(i,j)\\ &=\begin{cases}\max\left(\sum_{k=0}^{i}w_{(0,k)},\sum_{k=0}^{j}w_{(0,k)}\right),\text{if }\min(i,j)=0\\ \min\begin{Bmatrix}\mathrm{WLD}_{\alpha,\beta}(i-1,j)+w(\beta_j,0),\\ \mathrm{WLD}_{\alpha,\beta}(i,j-1)+w(0,\beta_j),\\ \mathrm{WLD}_{\alpha,\beta}(i-1,j-1)+w(\alpha_j,\beta_j)\end{Bmatrix},\text{otherwise}\end{cases}\end{aligned}\tag{2}$$

其中,$\mathrm{WLD}_{\alpha,\beta}(i,j)$ 表示属性序列 α 的前 i 位属性值转化为属性序列 β 下的前 j 位的加权编辑距离,$w(\alpha_j,\beta_j)$ 表示从属性值 α_i 替换成属性值 β_i 的加权距离. 该公式基于动态规划的思想,寻找最小的加权编辑距离,当 w 为 1 时,WLD 距离至少是两个属性序列长度的差值且不大于较大的那个属性的长度.

应用中要求的是一组顶点集合的属性相似度,给出多个顶点属性集合的最小编辑距离计算.

同一状态指所有的顶点属性序列相同,顶点属性集合的最小编辑距离是使集合中每一个属性序列,通过一系列编辑行为达到同一状态的最少编辑次数.

定理 2　同一状态的每个位置的属性值为其集合下该维度的最大频率值时,其编辑距离之和最小.

证明　设集合的最小编辑距离为 A,$A=\sum_{i}^{k}A_i$,A_i 为 i 维度上的编辑距离,设每个维度不同属性值距离为 1,权重为 1,n 为顶点数,k 为属性序列长度. 当同一状态在 i 维度上的属性值为该集合最大频率值时,设此时属性值最高频数为 m,则最少需要经过 $n-m$ 个编辑行为,该属性维度上所有属性序列相同,此时 $A_{i\min}=(n-m)\leqslant A_i$,同理 $\sum_{i}^{k}A_{i\min}\leqslant\sum_{i}^{k}A_i$,即集合的最小编辑距离 A 所生成的同一状态即为集合的每个维度属性频数最大的状态.　证毕

通常利用编辑距离刻画属性序列相似度计算:

$$\mathrm{Sim}(\alpha,\beta)=1-\frac{\mathrm{WLD}_{\alpha,\beta}}{\mathrm{Max}(\alpha,\beta)}\tag{3}$$

$\mathrm{Max}(\alpha,\beta)$ 表示属性序列 α,β 的最大长度,本文将顶点集合的属性序列通过编辑距离最小化时产生的编辑操作以编码进行映射和压缩.

图 2 中每一行代表顶点,纵轴表示属性维度,该属性精简自电影网络,A_1 的属性域是一个 0 ~ 1 值,A_2 属性域有 a、b、c,A_3 的属性域是连续变量,A_4 的属性域是一个集合类型,代表了该电影的主题;此时按照频率计数同一状态下的属性序列,在维度 A_1 上属性值 1 出现频率最大,则同一状态下 A_1 值为 1,由此计算同一状态属性序列为(1,b,10,action),并确保每个顶点到此状态的编辑距离之和最小. 通过构造编辑距离表生成右边的编辑行为列表,编辑行为列表表示最少经过以下操作,同一分组内所有顶点的属性序列将一样,当分组的属性相似度越高时,则编辑行为越少,编码代价越小.

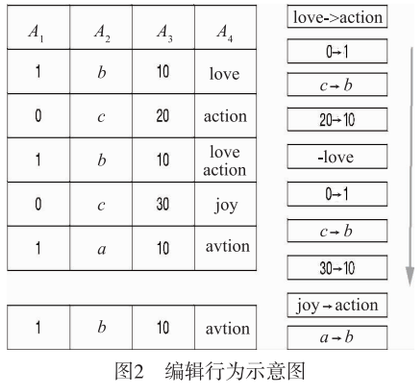

图2　编辑行为示意图

4.3.2　LZW 编码和解码

本文提出新的编码方式并借鉴 LZW 压缩思想,通过编辑行为码挖掘属性的二阶相似性. 编码主要包含编辑对象和编辑行为,编辑对象一般存储编辑的顶点或者属性信息.

图 3 描述编码规则,主要分为三段表示,其中第一段,第二段分别表示了顶点的 id 和属性的维度,第三段则是针对编辑行为的编码,例如“0001”则代表了编辑行为“love→action”,若后续单独出现“love→action”时以“0001”代替. 其中第二次出现“0→1”和“c→b”会生成复合码,以便压缩循环的编辑行为,编码过程中考虑权重的规则映射,例如在计算过程中“30→10”的编码代价应大于“20→10”的编码代价,若不兼顾概要图属性的还原,则只需要第三段编码计算属性编码代价.

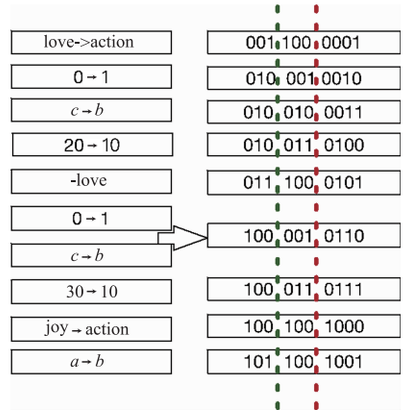

图3　LZW编码示意图

LZW 编解码算法流程如算法 1 所示. 依次读取编辑行为列表,将每一个编辑行为列表视为一个对象,P 存储上一个编辑行为,C 存储当前的编辑行为,D 为当前字典,存储了之前的编辑行为和编码的关系. 首先判断 PC 编辑行为是否针对同一顶点,若不是同一顶点,则在权重规则下输出编码,否则检测 PC 是否已属于字典,属于则将 PC 传递给 P,读入下一个编辑行为.

算法 1　LZW 编码

```
1:Input EB list,D=∅,
2:set P = last EB,C = next EB
3:for C in EB:
4:    if PC in D:
5:        P=PC,
6:        Continue
7:    else:
8:        Encode P
9:        Put PC in the D
10:       P=C
11:end for
```

综上,本文寻找顶点属性集合的最小编辑距离之和,将其转化为编辑行为列表,再结合改进的 LZW 压缩,将属性的一致性转化为编码,与结构相似性统一化为编码代价,给出属性代价一致性公式:

$$\mathrm{Cost}(u,v)_{\text{属性}}=\sum_{n\in(u,v)}\sum_{i=0}^{k}\mathrm{WLD}(a_i,a_{\max})*(L(D)+L(u,v)) \tag{4}$$

其中 $\mathrm{WLD}(a_i,a_{\max})$ 表示属性维度 i 上 a_i 与 $a_{\max}$ 的编辑距离,$L(D)$ 表示编码长度,$L(u,v)$ 表示存储 (u,v) 超点内部顶点 id 和属性的编码长度. $\mathrm{Cost}(u,v)_{\text{属性}}$ 表示属性一致性的编码代价.

5 概要模型

在属性图的结构一致性和属性一致性基础上,转化为编码存储.

5.1 代价函数

实际应用中不同的图数据,其顶点可能具有大量的邻居顶点少量的属性,导致了存储结构和属性的编码代价差距过大,在某些情况下用户有自己的偏好,比如更侧重于属性的均匀性,或者共同邻居的重要性. 权衡属性和结构,给出目标代价:

$$\mathrm{Cost}(u,v)=\gamma\,\mathrm{Cost}(u,v)_{\text{结构}}+(1-\gamma)\,\mathrm{Cost}(u,v)_{\text{属性}} \tag{5}$$

其中,$\mathrm{Cost}(u,v)$ 表示合并超点 (u,v) 所需要编码代价,期望 $\mathrm{Cost}(u)+\mathrm{Cost}(u)-\mathrm{Cost}(u,v)>0$ 的顶点对进行合并,合并后模型整体编码代价降低,减小规模,消除冗余信息,合并的优先级应由代价降低比率函数定义:

$$S(u,v)=\frac{\mathrm{Cost}(u)+\mathrm{Cost}(u)-\mathrm{Cost}(u,v)}{\mathrm{Cost}(u)+\mathrm{Cost}(u)} \tag{6}$$

当两个低度的等价顶点,合并时成本降低值小于相似的高度顶点对,但其完全等价顶点对应处在合并的第一优先级.

5.2 过滤机制

在部分图概要技术中,遍历图中所有的二跳顶点对,计算量是巨大的. 本文通过对二跳顶点对的合并成本进行预估,建立过滤机制,能过滤掉大部分无效二跳顶点对的计算.

过滤机制通过轻量级计算,判断二跳顶点对的合并代价,在上述讨论中,结构的编码代价,取决于公共邻居顶点是否大于非两者公共邻居顶点数,合并时的则需确保非公共邻居顶点小于公共邻居顶点. 属性编码代价,将问题转化为多个属性集合合并,寻找新的同一状态使编辑距离之和更小. 在属性维度 k 上,设合并 (u,v) 的属性最大频率值为 $a_{k_u\max}$,$a_{k_v\max}$,合并 w 后属性最大频率值为 $a_{k_w\max}$,若对每一个属性分布而言,均有 $a_{k_u\max}+a_{k_v\max}>a_{k_w\max}$,则编辑距离增大,编码代价增大,可过滤此二跳顶点对.

5.3 Greedy 策略

在顶点编码和相似性的度量基础上,采用最大堆结构,提出两种策略进行快速迭代合并.

Greedy 算法追求编码代价的最小化,全局优先合并最相似的顶点,寻找概要图最小表示,其迭代顺序遵循 $S(u,v)$ 的大小顺序. 如算法 2 所示.

算法 2 Greedy 算法(ASEC-Greedy 算法)

1:Input:$G=(V,E,A,F)$,$V_s=V_G$,$H=\varnothing$
2:for all pairs$(u,v)\in V_s$ that are 2 hops apart do:
3: if$(s(u,v)>0)$then insert $(u,v,s(u,v))$ into H;
4:end for
5:while $H=\varnothing$ do:
6: Choose pair$(u,v)\in H$ with the larget $s(u,v)$value;
7: $w=u\cup v$;
8: $V_s=V_s-\{u,v\}\cup\{w\}$;
9: for all $x\in V_s$ that are within 2 hops of u or v do:
10: Delete(u,x) and (v,x) from H;
11: if$(s(w,x)>0)$ then inser$(w,x,s(w,x))$into H;
12: end for
13: for all pairs(x,y),such that x or y is in N_w do:
14: Delete(x,y) from H;
15: if $(s(x,y)>0)$ then insert$(x,y,s(x,y))$into H;
16: end for
17:end while
18:Output:$E_s=D=\varnothing$;
19:for all pairs (u,v) such that $(u,v)\in V_s$ do:
20: if $|A_{u,v}|>(\Pi_{u,v}+1)/2$:
21: then add (u,v) to E_s;
22: end if
23:end for
24:return representation $R(G_s(V_s,E_s),D)$

5.4 Random 策略

Greedy 策略的迭代较慢,某些应用需要快速响应,帮助用户决策分析,因此提供一种 Random 策略,快速生成概要子图:

(1)随机选择顶点,选择其邻居顶点中降低成本代价比最大的顶点进行合并.

(2)针对属性的编辑距离的计算开销依然巨大,为此将稀疏的属性转化同一状态,减少大量的编辑行为.

Random 算法从邻域中选择最优顶点对进行合并,从而减少大量计算. 如算法 3 所示.

算法 3 Random 算法(ASEC-Random 算法)

1:Input:$G=(V,E,A,F)$,$V_s=V_G=V_t$;
2:while $V_t=\varnothing$ do:
3: Pick a node u randomly from V_t;
4: Find the node v with the largest value of $s(u,v)$ within 2 hops of u;

5:　if($s(u,v)>0$) then:
6:　　$w=u\cup v$;
7:　　$V_t=V_t-\{u,v\}\cup\{w\}$;
8:　　$V_s=V_s-\{u,v\}\cup\{w\}$;
9:　else:
10:　　Remove u from V_t;
11:　end if
12:end while
13:Output:$E_s=D=\varnothing$;
14:for all pairs (u,v) such that $t(u,v)\in V_s$ do
15:　if $|A_{u,v}|>(\Pi_{u,v}+1)/2$;
16:　　then add (u,v) to E_s;
17:　end if
18:end for
19:return representation $R(G_s(V_s,E_s),D)$

5.5　复杂度分析

ASEC 时间复杂度主要取决于算法的迭代过程,编码和 LZW 压缩均为并行. 分析 Greedy 算法迭代过程. 在一次超点 w 合并过程中,需要寻找其所有邻居顶点,包含这些邻居顶点的顶点对为 w 的三跳顶点数量,记为 3hop(w),此外,重新计算需要遍历两个顶点的所有边,在堆中的更新时间为 $O(\log|H|)$,顶点总数为 n,则堆的大小为 $|H|=n\times 2\text{hop}(w)$. 每一步迭代完的时间复杂度为:

$$O(4\text{hop}(w)+3\text{hop}(w)*(\log n+\log(2\text{hop}(w)))),$$

定义顶点的平均度为 d_v,则时间复杂度为 $O(d_v^3(d_v+\log n+\log d_v))$.

在 Random 算法中,只计算包含顶点 w 的所有邻居顶点对,顶点数为 2hop(w),整体代价和为 3hop(w),定义顶点平均复杂度为 d_v,则时间复杂度为 $O(d_v^3)$.

LZW 编码压缩的主要瓶颈在于编辑行为的检索,部分优化方案通过哈希表提升整体执行效率,但会带来双倍的空间开销,时间复杂度接近 $O(n)$.

6　实验

实验环境为 MacOS,RAM 16G,CPU 为 2.7GHz Intel Core i7,使用 java1.8,在表 1 所示 4 个真实数据集下展开实验,选取压缩率、属性熵以及时间代价进行对比. 压缩率用来衡量结构的紧密性,而熵指标用来衡量分组内属性的一致性,对比算法选择了 SAC[15]、k-SNAP[11]和 AGSUM[13]和 S-Entropy[14]四种算法.

SAC:一种针对属性图的聚类算法,通过随机游走距离度量顶点相似性.

k-SNAP:基于用户属性分组进行概要技术的经典方案;AGSUM 在 k-SNAP 基础上进一步优化了结构一致性.

S-Entropy:一种基于属性熵和结构熵的统一模型,产生近似同质分组.

表 1　实验数据集

Dataset	顶点/万	边/万	描述
Political Books	0.15	2	政治家的博客网络数据集,其属性描述了对相关政策的态度,如积极的态度或者消极.
DBLP	10	24	一个计算机类英文文献的数据库系统,顶点具备研究方向,性别,职称等属性.
MOVIE	6	20	豆瓣电影网络图,其中电影为顶点,若有共同主演形成链接关系,有票房,主题,评分等属性.
GTD	18	60	全球恐怖事件关系数据库,描述其恐怖事件的关联性,顶点具有袭击地点,目标,伤亡,宗旨等复杂属性.

6.1　性能分析

6.1.1　参数设置

本次实验首先针对该算法自身性能和相关参数进行分析,针对不同的参数,在 GTD 数据上采用两种算法进行概要和可视化分析,本文中所指压缩率越小,其占用存储空间越小.

参数 γ 控制着属性和结构的编码代价,图 4 显示了不同 γ 参数下的图数据压缩率. γ 越大则代表顶点的结构权重越大,侧重于结构相似性,其压缩率越小,这是因为存储属性图的大部分编码代价是由属性的校正引起的,由于 Greedy 算法追求全局的最紧密结构,所以压缩率小于 Random 算法.

图 5 和图 6 是 GTD 图数据在 $\gamma=0.3$ 和 $\gamma=0.6$ 时的 Greedy 算法概要图可视化,可见 γ 越大,其结构收缩的更加紧密,分组越少,其内部属性熵越大.

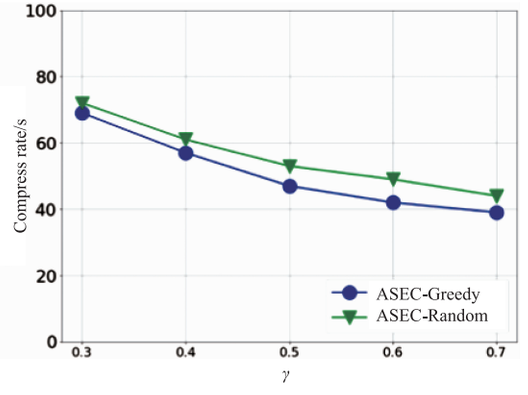

图4　不同的 γ 参数下的压缩率

6.1.2　Greedy 算法和 Random 算法

选取 MOVIE 构建同一数据集的不同规模,即将数据集构建顶点数为 1 万(MOVIE-1),2 万(MOVIE-2),4 万

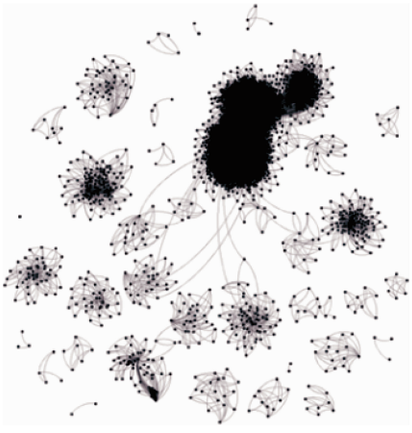

图5 γ=0.3时的可视化概要图

图6 γ=0.6时的可视化概要图

(MOVIE-3),8 万(MOVIE-4)的电影网络图,γ 设置为 0.8,分别从压缩率和时间上进行对比,进行性能分析.

图 7 显示两种算法在压缩率上的性能. 由于数据集自身的特性,其结构从稀疏转向稠密,整体的压缩率是减小的,即数据本身越紧密联系,压缩效果越好. 随着数据集的增大,Random 策略的压缩率越来越高,与 Greedy 算法差距越来越大.

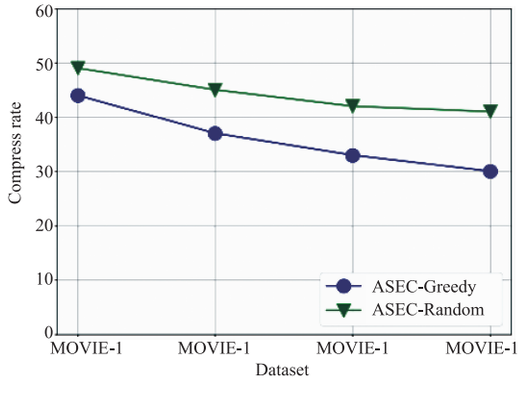

图7 Greedy和Random压缩率对比

图 8 显示 Greedy 策略的时间代价随着数据规模增大而显著提升,而 Random 策略依然保持着近似线性增长.

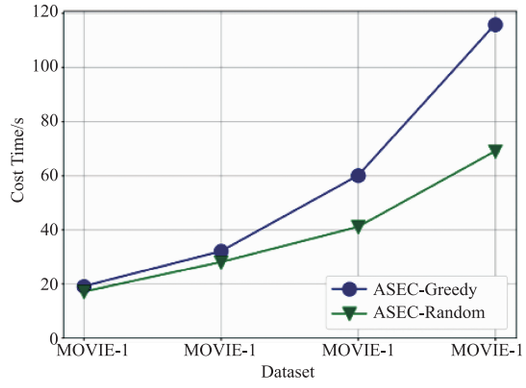

图8 Greedy策略和Random时间代价对比

6.1.3 模型构成

进一步理解模型的具体组成,采用 Greedy 策略进行概要分析,如图 9,D 表示字典编码代价,Gs 表示概要图的存储代价,CODE 表示编辑行为码的存储代价,Total表示代价之和. 由图可知,编辑行为码是最主要的存储开销,随着数据规模增大,其所占比例在降低,因为其大量的编辑行为码在 LZW 过程中被压缩为复合码. 所有的实验指标结果分析都跟数据图的属性成分和结构有着密切关联.

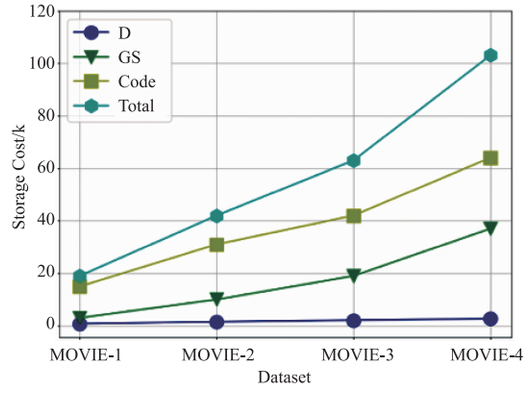

图9 概要模型的各部分编码代价

6.1.4 小结

实验表明:γ 参数控制着结构和属性相似性的权重,γ 越大,则概要图的结构更为紧凑,其压缩率越小. Greedy 算法每一次迭代选择全局最优顶点进行合并,并维护最大堆结构,适用于寻找最紧凑的概要图;Random 算法随机选择相邻顶点进行最优合并,减少了大量的计算开销,从模型组成分析来看,主要存储代价是属性的编辑行为码.

6.2　对比实验

本文采用 ASEC-Greedy 方法与现有的相关属性图概要技术进行对比实验,γ 设置为 0.6,主要从压缩率、熵、时间代价进行对比.

图 10 显示了在属性图上压缩率的实验结果,本文的 ASEC-Greedy 算法在压缩率上显示了优势.

在熵的指标上,如图 11 所示,k-SNAP 性能最好,在于仅考虑顶点的属性进行概要;AGSUM 考虑属性和结构的一致性,生成紧凑的概要图,压缩率降低;SAC 是基于距离的聚类方法,将属性转化为增广顶点,导致不同属性的顶点放在一起从而产生较高熵;S-Entropy 基于熵的维度平衡一致性. ASEC-Greedy 方法在熵的表现上优于 SAC 和 AGSUM,综合考虑了结构和属性的相似性,压缩率上性能最佳.

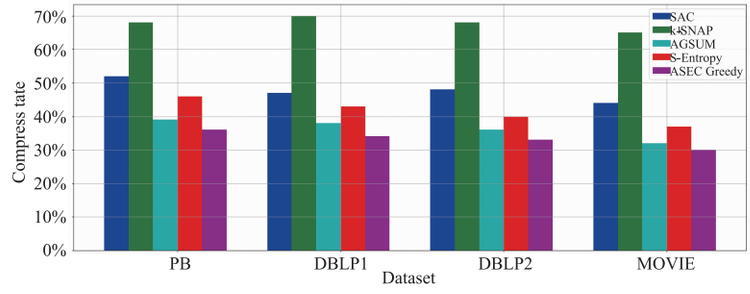

图10　属性图压缩率对比图

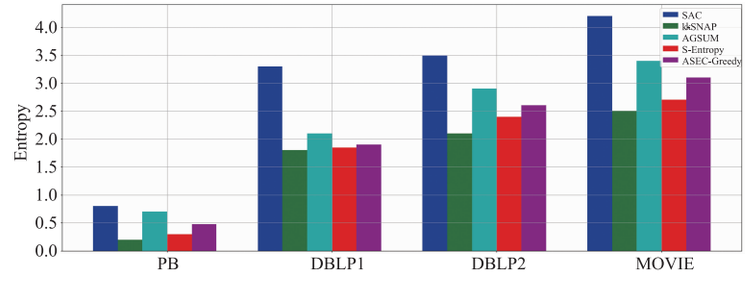

图11　属性图分组熵

图 12 是算法的运行时间,k-SNAP 时间开销最低,主要原因是其技术只考虑一个维度的概要化,计算量明显低于其他方法. 随着数据集复杂化,ASEC 性能更接近线性增长,表明了该技术在大型数据集上良好的扩展性.

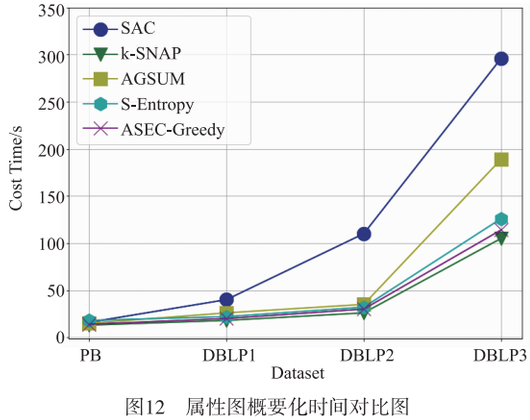

图12　属性图概要化时间对比图

6.3　重构

现有的属性图概要技术,均未涉及到原始图的重构,本文的概要模型,记录了编辑操作与编码映射,通过解码算法,完成原始图重构,并进行时间代价和信息保持率的实验. 如图 13 所示,由于 Greedy 策略对其属性和结构的最大程度的压缩,解码的时间代价小于 Random 策略. 重构的整体时间开销大于概要化,须按照编辑行为码单进程还原原始图.

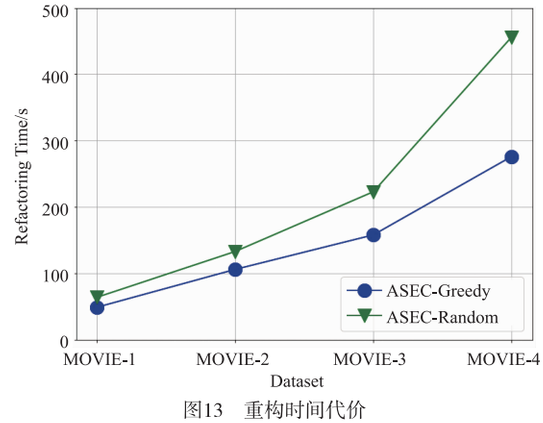

图13　重构时间代价

关于重构图的信息保持率(见图 14),编码是基于

无损压缩生成的,但解码过程中针对属性的编辑,会产生一定的误差,Greedy 算法的信息保持率始高于 Random 算法,其信息保持率均在 75% 以上.

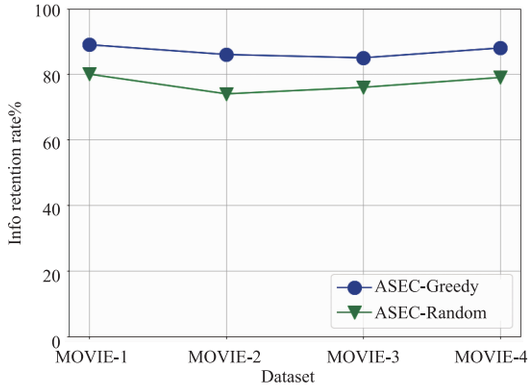

图14 信息保持率

6.4 可视化

图 15 为完整的恐怖袭击 GTD 图数据的可视化,以顶点表示恐怖袭击事件,边表示事件之间的关联性,包括恐袭组织、地域、时间上的关联,构成的图大约包含 20 万顶点,60 万条边,从原始图中很难挖掘出有价值的信息.

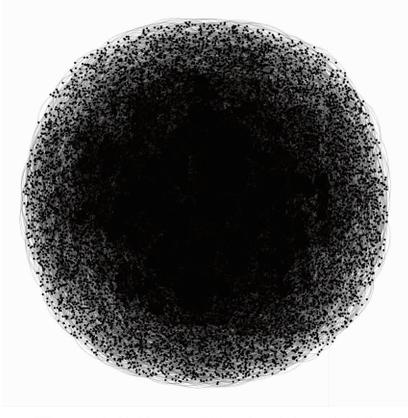

图15 完整的GTD恐怖袭击网络可视化

图 16 和图 17 为采用 Greedy 和 Random 算法的概要图可视化,超点的大小描述了超点内恐怖事件的数量. 概要图可以挖掘关键事件,即引起多个恐怖系列事件的导火索,例如链接多个超点的孤立顶点结构. 通过概要图寻找最邻近超点的最大重叠属性组,可以找出某恐怖事件的高度嫌疑犯罪组织.

图 16 展示了小型恐怖事件的频发演化成连环恐怖事件,部分顶点紧密连接依附一个超点. 图 17 描述了恐怖事件之间的相互作用关系,它们存在紧密的联系,但其袭击目标和方式的多样化,袭击意图和地理时间存在一定的相似性,这有利于挖掘隐藏的恐怖组织,对恐怖袭击的防范和预警工作存在着指导意义.

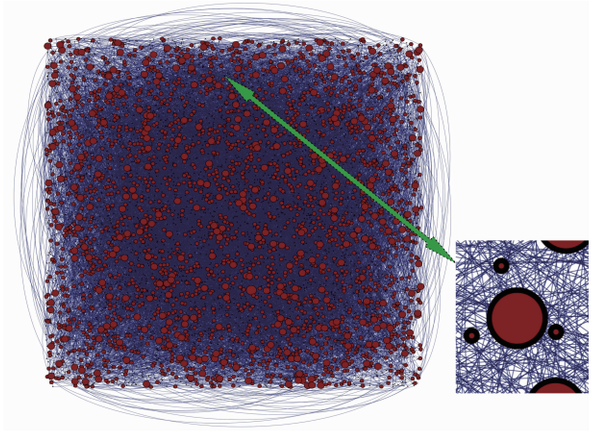

图16 ASEC-Greedy概要图可视化

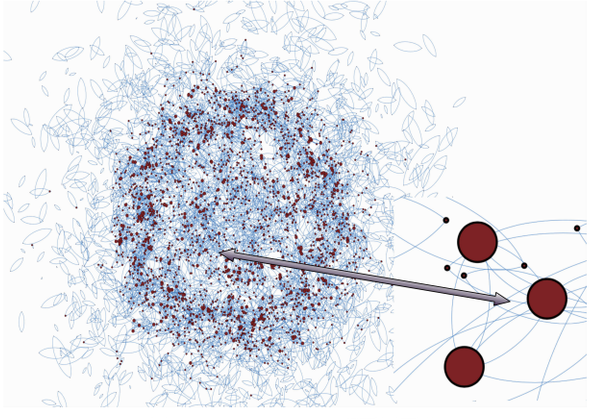

图17 ASEC-Random概要图可视化

7 总结

本文主要致力于研究属性图的概要工作,遵循 MDL 原则采用 WLD 距离和 LZW 压缩思想将属性和结构的一致性统一为编码代价,提出了新的属性图概要化模型,实现了 Greedy 算法和 Random 算法,迭代合并结构紧密,属性相似的顶点对,生成高质量的概要图,实验表明新提出的概要模型在结构和属性上均取得了良好的效果. 进一步的工作将集中在概要图的更新,如合并属性分组时编辑行为列表的动态更新,这将极大的减少算法的计算量,针对概要图的还原和编码的解码工作仍然有提升的空间,这将在以后的工作中进一步改进.

参考文献

[1] Liu Y, Safavi T, Dighe A, et al. Graph summarization methods and applications: A survey[J]. ACM Computing Surveys (CSUR), 2018, 51(3): 1 – 34.

[2] 王雄,董一鸿,施炜杰,等. 图概要技术研究进展[J]. 计

算机研究与发展,2019,56(6):1338 – 1355.
Wang Xiong, Dong Yihong, Shi Weijie, et al. Progress and challenges of graph summarization technique[J]. Journal of Computer Research and Development, 2019, 56(6): 1338 – 1355. (in Chinese)

[3] Beg M A, Ahmad M, Zaman A, et al. Scalable approximation algorithm for graph summarization[A]. Pacific-Asia Conference on Knowledge Discovery and Data Mining[C]. Cham: Springer, 2018. 502 – 514.

[4] 张建朋,陈鸿昶,王凯,等. 基于采样的大规模图聚类分析算法[J]. 电子学报,2018,47(8):1731 – 1737.
ZHANG Jian-peng, CHEN Hong-chang, WANG Kai, et al. A sampling-based graph clustering algorithm for large-scale networks[J]. Acta Electronica Sinica, 2019, 47(8): 1731 – 1737. (in Chinese)

[5] Kim J, Hwang I, Kim Y H, et al. Genetic approaches for graph partitioning: a survey[A]. Proceedings of the 13th Annual Conference on Genetic and Evolutionary Computation[C]. New York: ACM, 2011. 473 – 480.

[6] Beineke, Lowell W, Robin J Wilson, Peter J Cameron, et al. Topics in Algebraic Graph Theory[M]. UK: Cambridge University Press, 2004.

[7] Yang B, Cheung W K, Liu J. Community mining from signed social networks[J]. IEEE Transactions on Knowledge & Data Engineering, 2007, 19(10): 1333 – 1348.

[8] 刘露,胡封畔,牛亮,等. 异质网络中基于节点影响力的相似度度量方法[J]. 电子学报,2019,47(9):1929 – 1936.
LIU Lu, HU Feng-ye, NIU Liang, et al. Node influence based similarity measure method in heterogeneous network[J]. Acta Electronica Sinica, 2019, 47(9): 1929 – 1936. (in Chinese)

[9] Navlakha S, Rastogi R, Shrivastava N. Graph summarization with bounded error[A]. Proceedings of the 2008 ACM SIGMOD International Conference on Management of data[C]. USA: ACM, 2008. 419 – 432.

[10] Lim Y, Kang U, Faloutsos C. Slashburn: Graph compression and mining beyond caveman communities[J]. IEEE Transactions on Knowledge and Data Engineering, 2014, 26(12): 3077 – 3089.

[11] Tian Y, Hankins R A, Patel J M. Efficient aggregation for graph summarization[A]. Proceedings of the 2008 ACM SIGMOD International Conference on Management of data[C]. USA: ACM, 2008. 567 – 580.

[12] Zhang N, Tian Y, Patel J M. Discovery-driven graph summarization[A]. IEEE 26th International Conference on Data Engineering[C]. USA: IEEE, 2010. 880 – 891.

[13] Seo H, Park K, Han Y, et al. An effective graph summarization and compression technique for a large-scaled graph[J]. The Journal of Supercomputing, 2018, 10(12): 1 – 15.

[14] Liu Zheng, Yu Xu, Cheng Ho. Approximate homogeneous graph summarization[J]. Journal of Information Processing, 2012, 20(1): 77 – 88 .

[15] Zhou Y, Cheng H, Yu J X. Clustering large attributed graphs: An efficient incremental approach[A]. IEEE International Conference on Data Mining[C]. USA: IEEE, 2010. 689 – 698.

[16] Zneika M, Vodislav D, Kotzinos D. Quality metrics for RDF graph summarization[J]. Semantic Web, 2019, 10(3): 555 – 584.

[17] Gou X, Zou L, Zhao C, et al. Fast and accurate graph stream summarization[A]. IEEE 35th International Conference on Data Engineering[C]. USA: IEEE, 2019. 1118 – 1129.

[18] Qu Q, Liu S, Zhu F, et al. Efficient online summarization of large-scale dynamic networks[J]. IEEE Transactions on Knowledge and Data Engineering, 2016, 28(12): 3231 – 3245.

作者简介

王　雄　男,1994 年出生. CCF 学生成员,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、图数据挖掘、机器学习.

董一鸿(通信作者)　男,1969 年出生. 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