

Top-k 频繁子图挖掘的差分隐私保护算法

徐捷¹, 杨庚^{1,2}, 白云璐^{1,3}

- (1. 南京邮电大学 计算机学院, 江苏 南京 210046;
2. 江苏省大数据安全与智能处理重点实验室, 江苏 南京 210023;
3. 南京中医药大学 信息技术学院, 江苏 南京 210003)

摘要: 频繁子图挖掘是频繁模式挖掘的一种具体形式, 广泛应用于社会网络分析、生物技术、推荐系统等领域。然而, 图数据集中可能包含一些敏感的信息, 在挖掘过程中或发布频繁子图信息时都可能造成隐私的泄露。对此, 提出一种面向差分隐私保护的 top-k 子图挖掘算法——DP-TGM (Differential Private Top-k subGraph Mining)。算法首先依据挖掘出的频繁点和边对数据集剪枝, 然后将频繁的边依次进行扩展挖掘, 得到最终的 top-k 频繁子图。该算法使用一个优先队列存储临时挖掘到的前 k 个最频繁的子图, 在扩展挖掘的过程中不断更新队列里的元素, 并将阈值始终更新为队列里的最小噪音支持度, 减少图的扩展次数。算法使用拉普拉斯机制在三个不同的阶段对子图的真实支持度添加噪音, 并且采用均分法和特殊级数法对隐私预算进行合理的分配以提高数据可用性。文章用理论证明算法满足 ϵ -差分隐私保护, 且在不同规模的数据集上验证了算法的可用性。

关键词: top-k 频繁子图; 差分隐私; 拉普拉斯机制; 隐私预算; 数据可用性

中图分类号: TP309

文献标识码: A

文章编号: 1673-629X(2022)05-0080-07

doi: 10.3969/j.issn.1673-629X.2022.05.014

A Differential Privacy Protection Algorithm for Mining Top-k Frequent Subgraphs

XU Jie¹, YANG Geng^{1,2}, BAI Yun-lu^{1,3}

- (1. School of Computer Science, Nanjing University of Posts and Telecommunications, Nanjing 210046, China;
2. Jiangsu Province Key Laboratory of Big Data Security and Intelligent Processing, Nanjing 210023, China;
3. School of Information Technology, Nanjing University of Traditional Chinese Medicine, Nanjing 210003, China)

Abstract: Frequent subgraph mining is a specific form of frequent pattern mining, which is widely used in social network analysis, bio-technology, recommendation system and other fields. However, graph datasets may contain some sensitive information, which may lead to privacy leakage in the process of mining or publishing frequent subgraph information. To solve this problem, a DP-TGM (Differential Private Top-k subGraph Mining) is proposed. Firstly, the algorithm prunes the dataset according to the frequent vertices and edges, and then extends the frequent edges to get the final top-k frequent subgraph. The algorithm uses a priority queue to store the first k most frequent subgraphs which are temporarily mined. In the process of expanding mining, the elements in the queue are constantly updated, and the threshold is always updated to the minimum noise support in the queue, so as to reduce the number of graph expansion. The algorithm uses Laplacian mechanism to add noise to the true support of subgraphs in three different stages, and uses the averaging method and the special progression method to allocate the privacy budget reasonably to improve the data availability. It is proved theoretically that the algorithm satisfies the differential privacy protection, and the usability of the algorithm is verified on different data sets.

Key words: top-k frequent subgraphs; differential privacy; Laplacian mechanism; privacy budget; data usability

0 引言

由于在现实世界的应用程序中越来越多地使用图结构, 图挖掘已经变得非常流行。频繁子图挖掘是图挖掘中一个重要且有趣的问题, 其目标是提取给定数

据集中的出现次数高于指定阈值的子图^[1]。频繁子图挖掘的应用非常广泛, 推荐系统就是最常见的应用, 通过对浏览痕迹的挖掘, 推断用户的购买意向, 从而进行相关的推荐。同时, 在软件工程、生物化学、金融等领

收稿日期: 2021-06-03

修回日期: 2021-10-09

基金项目: 国家自然科学基金资助项目(61872197, 61972209)

作者简介: 徐捷(1997-), 男, 硕士, 研究方向为差分隐私保护; 杨庚, 教授, 硕/博导, 研究方向为隐私保护、云计算、访问控制。

域,频繁子图挖掘都有非常广阔的应用前景^[2-3]。

尽管频繁子图挖掘具有很高的实际应用价值,但是在挖掘和发布子图时都存在着隐私泄露的风险^[4]。假设有一个医疗保健的图数据库 D , D 中的每条记录表示一个用户的健康状况,子图表示某种疾病。对 D 进行频繁子图挖掘,挖掘结果显示了共有 10 个人患有肝炎。当新用户 Allen 的记录加入数据集 D 后,再次进行挖掘,肝炎患者的数量变为 11,就可以推断出 Allen 是肝炎患者,Allen 的隐私因此泄露。由此可见,频繁子图挖掘的结果不经过处理就发布很容易造成隐私的泄露,通过分析单个记录变化所引起的查询结果的变化,攻击者就可以轻易推断出用户的个人信息,这种攻击模式叫做差分攻击^[4]。

差分隐私保护技术有别于传统的隐私保护技术,以成熟的概率分布知识为基础,通过添加随机噪声扰动输出结果实现隐私保护,能够有效地抵御差分攻击。在差分隐私模型下,可以通过某种差分隐私随机算法 K 来发布数据,并为用户提供搜索界面,该算法保证了挖掘结果不会因为数据集中任一记录的改变而受到显著的影响^[5]。

目前,能够实现差分隐私保护的频繁子图挖掘算法并不多,而且这些算法难以同时满足高可用性与安全性,要么为了实现隐私保护,添加了过量的噪声,挖掘出的结果并不正确,数据的可用性大大降低;要么,隐私保护的力度不够,达不到 ϵ -差分隐私。为此,该文提出了一种满足差分隐私的 top-k 频繁子图挖掘算法 DP-TGM,其主要贡献如下:

(1) 为了实现差分隐私保护,在算法的三个阶段都使用拉普拉斯机制给予图的支持度添加噪声,将阈值也一直更新为队列中的最小噪声支持度。

(2) 为了提高数据可用性,采用均分法和特殊级数法来分配隐私预算,降低误差;同时,不断更新变大的阈值减少了子图的扩展比较次数,进一步提升准确性。

(3) 理论证明,算法实现了差分隐私保护;通过对比实验,在不同规模的真实图数据集上,DP-TGM 算法都展现出了更高的数据可用性。

1 相关工作

近些年来,已经有不少学者提出了满足差分隐私的 top-k 频繁模式挖掘算法。2012 年, Li 等人提出了 PrivBasis 算法^[6],该算法通过将输入数据投影到人们所关心的少数选定维度上来应对高维性的挑战。PrivBasis 算法在高维数据集上的表现远优于 TF 算法,但其可用性却因随机截断引起的截断误差而大受影响。针对此问题,蒋辰等人^[7]提出了 TrunSuper 算

法,该算法使用新的截断方法,将事务中支持度较小的项剔除,通过降维减小项集的支持度误差。

在满足差分隐私的 top-k 子图挖掘方面,2013 年, Shen 等人^[8]提出了一种基于采样的频繁子图挖掘算法 Diff-FPM,该算法可概括为以下两个步骤:

(1) 采样:使用马尔可夫链蒙特卡罗抽样 (MCMC) 的方法来扩展指数机制,并使用扩展指数机制直接从图数据集中选择频繁子图,重复此过程,直到获得了 top-k 频繁子图;

(2) 扰动:对获得的 top-k 子图的支持度添加符合拉普拉斯分布的噪音。

然而,当样本的分布不可观察时,验证 MCMC 的收敛仍然是一个悬而未决的问题,Diff-FPM 算法仅仅满足较弱的 (ϵ, δ) -差分隐私。此外,算法从所有子图构成的空间中进行挑选,引入的噪音过大,数据的可用性较差。

张啸剑等人^[9]提出的 DP-tokP 算法在差分隐私保护的模型下进行 top-k 频繁模式的挖掘,可用于 top-k 子图挖掘,其思想如下:

(1) 挖掘出图数据集中所有支持度大于阈值的频繁子图,存入集合 S 中;

(2) 设置打分函数为子图的支持度,为 S 中的每个子图进行打分。使用指数机制为每个子图按照分值赋予权重,并降序排列,从排列好的子图中不放回地抽取 k 个子图,形成最终的 top-k 子图集合;

(3) 为挖掘出的 top-k 子图的支持度添加拉普拉斯噪音。

该算法同样使用了差分隐私的两种机制,满足 ϵ -差分隐私保护。但是,在使用指数机制挑选子图之前,需要挖掘出所有的候选集,初始阈值的选择不同,候选集的个数就不同,候选集越多,产生的扰动越大,挖掘结果的准确性就越低。

以上算法在兼顾安全性和数据可用性方面依旧有很大的提升空间,难以运用于实际应用。为此,该文提出了 DP-TGM 算法,采用合理的隐私预算分配方法,提高数据可用性,在挖掘过程和发布数据时达到隐私保护的效果。

2 理论基础

2.1 差分隐私

定义 1 近邻数据集。给定两个数据集 D 和 D' , 当且仅当两数据集之间只相差一条记录时,可称之为近邻数据集^[10]。

定义 2 ϵ -差分隐私。给定两个近邻数据集 D 和 D' , 若算法 A 作用在数据集 D 和 D' 上的结果满足不等式(1),则称算法满足 ϵ -差分隐私。

$$\Pr[A(D) = 0] \leq \exp(\varepsilon) \times \Pr[A(D') = 0] \quad (1)$$

不等式中, $\Pr[X]$ 是事件 X 发生的概率, 即隐私泄露的概率, 它由算法 A 的随机属性确定。参数 ε 是隐私保护预算, ε 值与隐私保护程度成反比, ε 越小, 两个近邻数据集的相同输出的概率越接近, 隐私保护的度越高。

定义 3 全局敏感度。对于任意一个函数 $f: D \rightarrow R^n$, 它的全局敏感性 Δf 定义为:

$$\Delta f = \max_{D, D'} \|f(D) - f(D')\|_1 \quad (2)$$

R 表示所映射的实数空间, n 表示函数 f 的查询维度。全局敏感度的大小与具体的数据集无关, 由查询函数决定, 反映了函数 f 在 D 和 D' 上变化的最大范围^[11]。

添加噪声是使算法实现差分隐私保护的主要途径, 最常用的噪声添加机制是拉普拉斯 (Laplace) 机制和指数机制, 其中, 拉普拉斯机制主要适用于数值型输出, 而指数机制适用于非数值型输出^[12]。

定义 4 拉普拉斯机制。拉普拉斯机制通过在查询结果上添加满足 Laplace 分布的噪音来实现隐私保护。给定数据集 D , 若有函数 $f: D \rightarrow R^d$, 其敏感度为 Δf , 当算法 A 的输出结果满足下列等式:

$$A(D) = f(D) + \langle \text{Lap}_1(\Delta f/\varepsilon), \dots, \text{Lap}_n(\Delta f/\varepsilon) \rangle \quad (3)$$

则算法 A 满足 ε -差分隐私, 其中, $\text{Lap}_i(\Delta f/\varepsilon)$ ($1 \leq i \leq n$) 是相互独立的拉普拉斯变量。噪声大小与 Δf 成正比, 与 ε 成反比。

定义 5 指数机制。指数机制首先需要制定一个打分函数 $u: (D \times O) \rightarrow R$, 设 A 为指数机制下的某个算法, 则输出结果为:

$$A(D, u) = \{ r \mid \Pr[r \in O] \propto \exp\left(\frac{\varepsilon u(D, r)}{2\Delta f}\right) \} \quad (4)$$

由公式 (4) 可知, 分值越高, 添加的噪声就越大, 被输出的概率也越大。

定义 6 差分隐私的序列组合性^[13]。假设算法 A 可分解为 n 个过程 $A_i (i \in [1, n])$, 且均作用在同一数据集 D 上, 若 A_i 满足 ε_i -差分隐私, 则算法 A 满足 ε -差分隐私, 其中 $\varepsilon = \sum_{i=1}^n \varepsilon_i$ 。该性质表明: 当多个差分隐私保护算法作用于同一数据集时, 整体的隐私预算为各个算法隐私预算的总和, 所以也可称作串型组合特性。

2.2 频繁子图挖掘

频繁子图挖掘是频繁模式挖掘的问题之一, 其任务是找出图数据库中频繁出现的子图结构。

定义 7 支持度。子图 g 的支持度指的是图数据库中包含子图 g 的记录个数, 同一记录中出现多次

也只记一次, 文中支持度用 $\text{Sup}(g)$ 表示。

定义 8 噪音支持度^[12]。某一子图 g 在其支持度 $\text{Sup}(g)$ 上添加噪音形成的支持度称为噪音支持度, 如式 (5) 所示:

$$\text{NSup}(g) = \text{Sup}(g) + \text{noise} \quad (5)$$

其中, $\text{NSup}(g)$ 表示子图 g 的噪音支持度, noise 表示添加的噪音。

定义 9 频繁子图挖掘。给定图数据库 $D = \{G_1, G_2, \dots, G_n\}$ 和阈值 θ , 支持度大于等于 θ 的子图被称为频繁子图, 频繁子图挖掘就是找出图数据库中所有的频繁子图。

定义 10 top-k 频繁子图挖掘^[14]。给定图数据库 $D = \{G_1, G_2, \dots, G_n\}$ 和用户自定义值 k (一般来说, k 的取值较低), top-k 频繁子图挖掘就是找出图数据库中支持度排名前 k 的频繁子图。

3 DP-TGM 算法

DP-TGM 算法的挖掘对象是图数据集中支持度排名前 k 的子图, 算法共分为三个阶段: 预处理阶段、深度挖掘阶段和噪音添加阶段。

3.1 DP-TGM 算法概述

算法使用两个优先级队列 Q_k 和 Q_s , Q_k 用来存储临时挖掘到的 top-k 频繁子图, 噪音支持度小的优先级高; Q_s 用来存储待拓展的频繁边, 支持度高的优先级高。算法将隐私预算 ε 分为三份, 分别用于预处理、深度挖掘和噪音添加阶段, 算法 1 展示了算法的整体框架。

算法 1: DP-TGM 算法。

输入: 图数据集 GD ; 隐私预算 ε ; k 。

输出: top-k 频繁子图及其噪音支持度。

1. Initialize the priority queue Q_k ;
2. Initialize the priority queue Q_s ;
3. $\varepsilon = \varepsilon_1 + \varepsilon_2 + \varepsilon_3$;
4. Pre-mining(GD, ε_1);
5. Top-k-mining(GD, ε_2, Q_s);
6. Add-noise(Q_k, ε_3);
7. return Q_k 。

如算法 1 所示: 给定图数据集 D , 隐私预算 ε 和 k , 算法通过三个阶段的挖掘处理, 最终得到 top-k 频繁子图集合 Q_k 。其中, 预处理阶段主要用来获取频繁点和频繁边, 分配隐私预算 ε_1 ; 深度挖掘阶段将预处理阶段所得的频繁边进行深度挖掘, 得到最终的 top-k 子图集合, 分配隐私预算 ε_2 ; 噪音添加阶段对挖掘结果的支持度添加拉普拉斯噪音进行扰动, 分配隐私预算 ε_3 。

下面将对三个阶段展开讲解。

3.2 预处理阶段

预处理阶段是 DP-TGM 算法的第一阶段,将图数据集进行遍历,获得频繁点和频繁边,更新 Q_k 和 Q_s 并得到新的阈值。预处理阶段的算法如下。

算法 2:预处理算法 Pre-mining(GD, ε_1)。

输入:图数据集 GD ;隐私预算 ε_1 。

输出:预处理阶段的 top-k 子图集合 Q_k, Q_s 。

```

1.  $\theta = 10$ ;
2. 挖掘出频繁的点 and 边,并存入  $Q_k$  中;
3. if ( $|Q_k| > k$ ) //  $|Q_k|$  表示队列的大小
4.   删去  $Q_k$  中支持度较低的子图;
5. end if
6. 依据  $Q_k$  对数据集  $GD$  剪枝;
7. 将  $Q_k$  中的频繁边复制并存入  $Q_s$  中;
8. for each subgraph  $s$  in  $Q_k$ :
9.    $Nsup(s) = sup(s) + Laplace(|Q_k| / \varepsilon_1)$ ;
10. End for
11.  $\theta = Nsup(Q_k. peek)$ ; // 阈值更新为  $Q_k$  中的最小噪音支持度;
12. return  $Q_k, Q_s$ 。
```

算法 2 是对预处理阶段的具体描述,首先设置一个较低的阈值 θ (例如 $\theta = 10$),遍历数据集 GD ,挖掘出支持度大于阈值的顶点和边,存入 Q_k 中。若是 Q_k 中的子图个数大于 k ,则将支持度低的子图移出队列,依据 Q_k 进行剪枝。 Q_s 中放入 Q_k 中的频繁边,用来进行下一轮的拓展。最后将 Q_k 中的子图添加上拉普拉斯噪音,更新阈值为 Q_k 中噪音支持度的最小值。

定理 1:算法 2 满足 ε_1 - 差分隐私。

证明:在图数据集 GD 中,添加或删除一条记录,对每个子图的支持度影响最多为 1,所以算法 2 的敏感度为 1。因此,如算法 2 的第 7 行所示,给 Q_k 中的每个子图添加的噪音为 $Laplace(|Q_k| / \varepsilon_1)$ ($|Q_k|$ 表示队列 Q_k 的大小),则每个子图都满足 $\varepsilon_1 / |Q_k|$ - 差分隐私,由定义 6 可得,算法 2 满足 ε_1 - 差分隐私。

证毕。

3.3 深度挖掘阶段

DP-TGM 算法在深度挖掘阶段有两大重要思想:

(1) 不断更新阈值。在深度挖掘阶段,阈值 θ 随着优先权队列 Q_k 的变化而不断改变,始终将其更新为 Q_k 中的最小噪音支持度,这样可以有效地提高挖掘效率。

(2) 合理分配隐私预算。隐私预算的分配涉及到噪音添加的强度,也直接影响到数据的可用性与安全性。在深度挖掘阶段,将会使用两种隐私预算分配方法:均分法和特殊级数法,对 ε_2 进行两次分配,以更低的速率释放隐私预算,提高挖掘结果的准确性。

算法 3:深度挖掘算法 Top-k-mining(GD, ε_2, Q_s)。

输入:图数据集 GD ;隐私预算 ε_2 ; k ;算法 1 得到的

Q_k, Q_s 。

输出:top-k 子图集合 Q_k 。

```

1.  $\varepsilon_a = \varepsilon_2 / |Q_s|$ ;
2. while  $Q_s$  is not empty:
3.    $g \leftarrow \text{pop the subgraph with highest priority in } Q_s$ ;
4.   if  $Q_k$  contains  $g$  then
5.     初始化优先权队列  $Q$ ; // 支持度越高,则优先级越高;
6.   对  $g$  扩展,并将扩展的图存入  $Q$ ;
7.   for each  $g$  in  $Q$ :
8.      $\varepsilon_i = \frac{\varepsilon_a}{i(i+1)}$ ;
9.      $Nsup(i) = Nsup(i) + Laplace(1/\varepsilon_i)$ ;
10.    if ( $\min(g) \ \&\& \ Nsup(g) > \theta$ )
11.      store  $g$  into  $Q_k$ ;
12.    if  $|Q_k| > k$  then
13.       $Q_k \cdot \text{pop}()$ ;
14.       $\theta = Nsup(Q_k \cdot \text{peek})$ ;
15.    end if
16.  else
17.    break;
18.  end for
19.  else
20.    break; // 算法结束
21. end while
22. return  $Q_k$ 。
```

算法 3 首先将隐私预算等分为 $\varepsilon_a = \varepsilon_2 / |Q_s|$,将 Q_s 中优先级最高(支持度最高)的子图 g 弹出。如果 Q_k 不包含 g ,则算法终止,因为 g 是待拓展的子图里支持度最高的,而 g 被 Q_k 剔除,说明其支持度不够。如果 Q_k 中包含 g ,则将 g 按照最右路径规则进行拓展,并将拓展的图放入优先权队列 Q 中。将 Q 中的每个图 s 按特殊级数法进行隐私预算的分配,添加拉普拉斯噪音,如果噪音支持度大于 θ ,则将 s 插入 Q_k 中,然后判断 Q_k 的大小,按情况对 Q_k 进行更新,且将支持度始终设置为 Q_k 中的最小噪音支持度。若是 s 的噪音支持度小于阈值,则关于 g 的扩展结束。当 Q_s 为空时或 Q_s 中优先级最高的边都不满足要求,则算法结束,此时, Q_k 存储的就是最终的 top-k 频繁子图。

定理 2:差分隐私保护方法中,特殊级数法^[15]采用如下预算分配方式:

$$\varepsilon_i = \frac{\varepsilon}{i(i+1)} \quad (6)$$

则有限次隐私预算分配满足 ε - 差分隐私。

证明:因为任意一次隐私预算分配量 $\varepsilon_i > 0$, ($i \in \mathbb{N}^+$),有限次 ($n < \infty$) 分配的隐私预算分配量之和为:

$$\sum_{i=1}^n \frac{\varepsilon}{i(i+1)} < \sum_{i=1}^{\infty} \frac{\varepsilon}{i(i+1)} = \varepsilon \quad (7)$$

则有限次隐私预算分配 ε - 满足差分隐私。

证毕。

定理 3: 算法 3 满足 ε_2 - 差分隐私。

证明: 算法 3 首先使用均分法将 ε_2 分为 $\varepsilon_a = \varepsilon_2 / |Q_s|$, Q_s 中的每个子图得到了 ε_a 的隐私预算。算法依次将 Q_s 中优先级最高的子图 g 移出队列, 进行深度挖掘, 使用特殊级数法进行隐私预算分配, 由定理 2 可得, 每个子图 g 的深度挖掘满足 ε_a - 差分隐私。又由定义 6 可得, 整个算法 2 满足 $\varepsilon_a \times |Q_s| = \varepsilon_2$ - 差分隐私保护。

证毕。

3.4 噪音添加阶段

噪音添加阶段是 DP-TGM 算法的最后一个阶段, 主要思想是对挖掘出的 top-k 频繁子图的真实支持度添加拉普拉斯噪声进行扰动。

算法 4: 噪音添加算法 Add-noise(Q_k, ε_3)。

输入: 算法 2 得到的 Q_k , 隐私预算 ε_3 。

输出: 加噪后的 top-k 子图集合 Q_k 。

1. for subgraph G_i in Q_k :
2. $\varepsilon_i = \varepsilon_3 / k$;
3. $\text{Nsup}(G_i) = \text{sup}(G_i) + \text{Laplace}(\varepsilon_i)$;
4. end for
5. Return Q_k 。

定理 4: 算法 4 满足 ε_3 - 差分隐私。

证明: 这里使用均分法将隐私预算均分为 k 份, Q_k 中的每个频繁子图添加的噪音为 $\text{Laplace}(k/\varepsilon_3)$ 。算法 4 和算法 2 一样使用均分法来分配隐私预算, 所以同理可证, 算法 4 满足 ε_3 - 差分隐私。

证毕。

3.5 DP-TGM 算法隐私性证明

定理 5: DP-TGM 算法满足 ε - 差分隐私保护。

证明: DP-TGM 算法将隐私预算分为三份, 分别用于三个阶段: 预处理 (ε_1)、top-k 子图挖掘 (ε_2)、噪音添加 (ε_3)。该文设定隐私预算的分配比例 $\varepsilon_1 : \varepsilon_2 : \varepsilon_3 = 1 : 5 : 4$ 。由定理 1 可得, 算法 1 满足 ε_1 - 差分隐私, 由定理 3 可得, 算法 2 满足 ε_2 - 差分隐私, 由定理 4 可得, 算法 3 满足 ε_3 - 差分隐私。由定义 6 差分隐私的序列组合性可得, 算法满足 $(\varepsilon_1 + \varepsilon_2 + \varepsilon_3)$ - 差分隐私保护, 而 DP-FGM 算法的整体隐私预算 $\varepsilon = \varepsilon_1 + \varepsilon_2 + \varepsilon_3$, 所以 DP-FGM 算法满足 ε - 差分隐私保护。

证毕。

4 实验结果与分析

本节将通过对比实验来验证算法的数据可用性。实验环境为 Inter(R) Core(TM) i5-8250U CPU @ 1.60 GHz, 8.00 GB 内存 Windows10 64 位操作系统。

实验将在三个真实的图数据集上进行测试, 分别是 NCI1、Protein 和 Reddit-multi。NCI1 是抗非小细胞

肺癌和卵巢癌细胞系活性筛选的化合物数据集, Protein 是蛋白质分子结构数据集, Reddit-multi 则是社交网络数据集。表 1 展示了各数据集的具体特征, 包括图的个数 (graph count), 平均节点数 (avg-nodes) 和平均边数 (avg-edges)。

表 1 图数据集信息

dataset	graph count	avg-nodes	avg-edges
NCI1	4 172	29.87	32.30
Protein	10 023	30.62	411.28
Reddit-multi	11 929	391.41	456.89

该文将 DP-TGM 算法与 DP-tokP 算法和 Diff-FPM 算法进行比对, 实验所涉及的代码均由 java 语言实现。由于噪声的加入, 数据具有随机性, 实验结果存在不确定性, 因此采取多次试验取平均值的方式来记录结果。

4.1 度量指标

该文使用两个度量标准: F1-Score 和 RE。F1-Score 主要衡量挖掘的频繁子图结果的可用性, RE 则用来衡量子图支持度的准确性。F1-Score 的比较结果使用条形图展示, 而 RE 的比较结果用折线图展示。

定义 11 F1-Score^[16]:

$$\text{F1-score} = \frac{\text{Accuracy} \times \text{Recall}}{\text{Accuracy} + \text{Recall}} \times 2 \quad (8)$$

其中, Accuracy 表示精确率, Recall 表示召回率。Accuracy = $(\text{Up} \cap \text{Us}) / \text{Up}$, Recall = $(\text{Up} \cap \text{Us}) / \text{Us}$, Up 是在差分隐私下进行频繁子图挖掘的结果, Us 则是频繁子图挖掘的准确结果。F1-Score 将精确率和召回率综合考量, 取值区间为 $[0, 1]$, F1-Score 的值越大, 则代表数据效用越好。

定义 12 RE (相对错误率):

$$\text{RE} = \frac{\sum_{i=0}^{k-1} \frac{|\text{Nsup}(i) - \text{Sup}(i)|}{\text{Sup}(i)}}{k} \quad (9)$$

式中, $\text{Nsup}(i)$ ($0 \leq i \leq k-1$) 是结果集中第 i 个子图的噪音支持度, $\text{Sup}(i)$ 则是第 i 个子图的真实支持度。RE 用来衡量挖掘到的频繁子图的支持度的错误率, 取值区间为 $[0, \infty]$ 。RE 的值与引入的噪音量大小相关, 噪音越大, 噪音支持度与真实支持度的差值越大, RE 的值就越高, 数据可用性就越差。所以, RE 的值越小, 算法的数据效用越高。

4.2 可用性随 k 取值的变化

DP-TGM 算法是进行 top-k 频繁子图挖掘的算法, k 的大小将影响隐私预算的分配。实验通过调整 k 的大小来对比三个算法的 F1-Score 和 RE 值。这里, 统一设置隐私预算 ε 为 1, k 从 30 变化到 150, 间隔为

30,在三个数据集上进行测试,结果如图1~图3所示。随着 k 的变大,三个算法的F1-Score都在降低,而RE值在增大,说明随着 k 增大,算法的数据效用在降低。而当 k 相同时,DP-TGM算法的F1-Score始终要比DP-tokP算法和Diff-FPM算法高,而RE要比它们低,验证了DP-TGM算法的优越性。

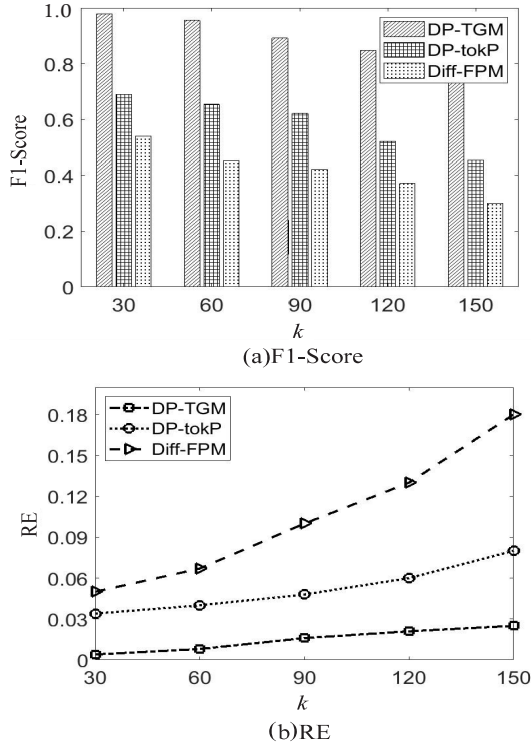


图1 数据集NCII随 k 变化时可用性变化情况

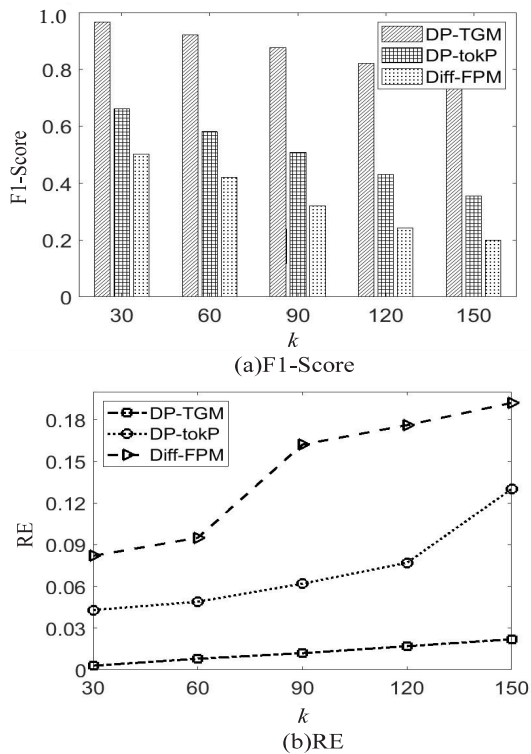


图2 数据集Protein随 k 变化时可用性变化情况

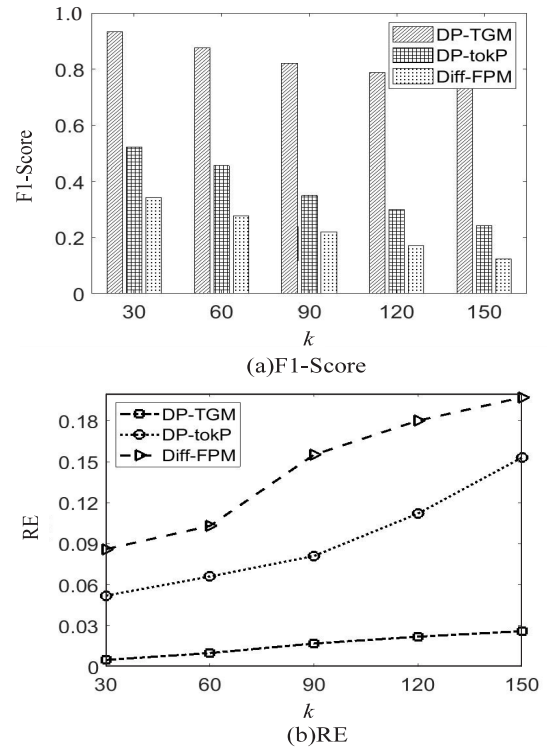


图3 数据集Reddit-multi随 k 变化时可用性变化情况

4.3 可用性随隐私预算 ϵ 取值的变化

隐私预算 ϵ 的取值可以影响到噪声的大小,而每个算法隐私预算的分配也不相同,所以 ϵ 的变化会影响挖掘结果的数据效用。这里统一设置 k 为50, ϵ 从0.5变化到1.5,间隔为0.25,在NCII和Protein两个数据集上进行测试,实验结果如图4、图5所示。

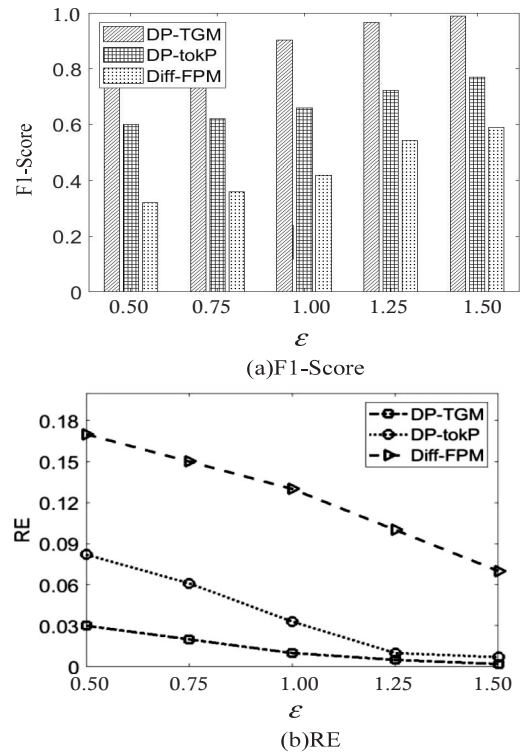


图4 数据集NCII随 ϵ 变化时可用性变化情况

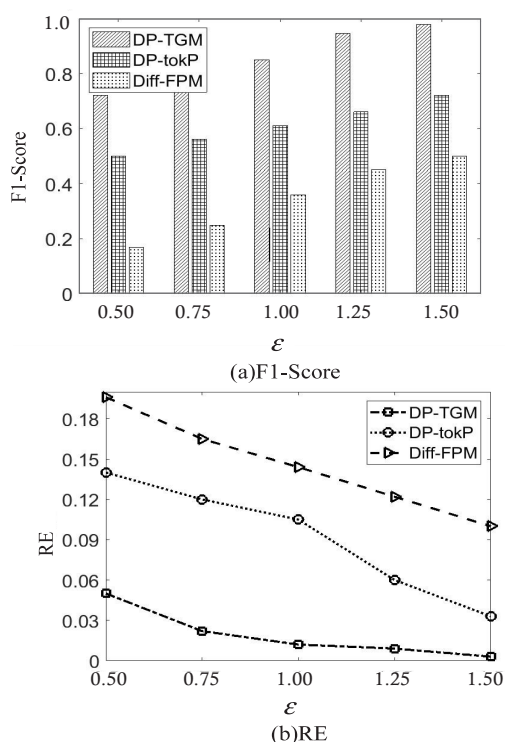


图 5 数据集 Protein 随 ϵ 变化时可用性变化情况

随着 ϵ 的增大,三种算法的 F1-Score 值都在增大,而 RE 值在减小,这是因为 ϵ 的增大,引入的噪音减小,提高了数据效用。而在 ϵ 从 0.5 变化到 1.5 的过程中,DP-TGM 算法的 F1-Score 值始终最高,RE 值始终最低,再次验证了文中算法的优越性。

5 结束语

设计并实现了一个满足差分隐私的 top-k 子图挖掘算法 DP-TGM,通过不断地更新优先权队列 Q_k ,将不满足要求的子图剔除,同时更新阈值,提高挖掘的效率。为了提高数据的可用性,使用均分法和特殊级数法进行隐私预算的分配,以更低的速度释放隐私预算。同时,在不同规模的真实图数据集上的测试成果也显示了算法具有更高的数据可用性。为了提高挖掘性能和结果的准确性,在挖掘子图的过程中浪费了一些隐私预算,因此下一步将研究如何减少隐私预算的浪费。

参考文献:

- [1] CHENG X, SU S, XU S, et al. A two-phase algorithm for differentially private frequent subgraph mining [J]. IEEE Transactions on Knowledge and Data Engineering, 2018, 30 (8): 1411-1425.
- [2] SU S, XU S, CHENG X, et al. Differentially private frequent itemset mining via transaction splitting [J]. IEEE Transactions on Knowledge & Data Engineering, 2015, 27 (7): 1875-1891.
- [3] SU Xin, FAN Kuan, SHI Wenbo. Privacy-preserving distributed data fusion based on attribute protection [J]. IEEE Transactions on Industrial Informatics, 2019, 15 (10): 5765-5777.
- [4] SORIA-COMAS J, DOMINGO-FERRER J, SÁNCHEZ D, et al. Individual differential privacy: a utility-preserving formulation of differential privacy guarantees [J]. IEEE Transactions on Information Forensics and Security, 2017, 12 (6): 1418-1429.
- [5] ZHU T, LI G, ZHOU W, et al. Differentially private data publishing and analysis: a survey [J]. IEEE Transactions on Knowledge & Data Engineering, 2017, 29 (8): 1619-1638.
- [6] LI Ninghui, QARDAJI W, SU Dong, et al. PrivBasis: frequent itemset mining with differential privacy [J]. Proc of the VLDB Endowment, 2012, 5 (11): 1340-1351.
- [7] 蒋辰, 杨庚, 白云璐, 等. 面向隐私保护的频繁项集挖掘算法 [J]. 信息安全, 2019 (4): 73-81.
- [8] SHEN E, YU T. Mining frequent graph patterns with differential privacy [C] // Proceedings of the ACM SIGKDD international conference on knowledge discovery and data mining. [s. l.]: ACM, 2013: 545-553.
- [9] 张啸剑, 王森, 孟小峰. 差分隐私保护下一种精确挖掘 top-k 频繁模式方法 [J]. 计算机研究与发展, 2014, 51 (1): 104-114.
- [10] 唐海霞, 杨庚, 白云璐. 自适应差分隐私预算分配策略的直方图发布算法 [J]. 计算机应用研究, 2020, 37 (7): 1952-1957.
- [11] ZHU T, XIONG P, LI G, et al. Correlated differential privacy: hiding information in non-IID data set [J]. IEEE Transactions on Information Forensics and Security, 2015, 10 (2): 229-242.
- [12] KARTAL H B, LIU Xiaoping, LI Xiaobai. Differential privacy for the vast majority [J]. ACM Trans on Management Information Systems, 2019, 10 (2): 8. 1-8. 15.
- [13] KUMAR T, KATEBI S, DHIFLI W, et al. Discovery of functional motifs from the interface region of oligomeric proteins using frequent subgraph mining [J]. IEEE/ACM Transactions on Computational Biology and Bioinformatics, 2019, 16 (5): 1537-1549.
- [14] ABDELHAMID E, CANIM M, SADOOGHI M, et al. Incremental frequent subgraph mining on large evolving graphs [J]. IEEE Transactions on Knowledge and Data Engineering, 2017, 29 (12): 2719-2723.
- [15] 王璇. 差分隐私保护中隐私预算的优化与应用 [D]. 南京: 南京邮电大学, 2019.
- [16] 丁丽萍, 卢国庆. 面向频繁模式挖掘的差分隐私保护研究综述 [J]. 通信学报, 2014, 35 (10): 200-209.