南京邮电大学

毕 业 设 计（论 文）

题　　目	基于社交网络图结构的差分隐私发布模型研究
专　　业	信息科技英才班（信息安全）
学生姓名	顾婕
班级学号	Q14010107
指导教师	黄海平
指导单位	计算机学院、软件学院、网络空间安全学院

日期：　2017 年 12 月　15　日至　2018 年　6 月　8　日

毕业设计（论文）原创性声明

本人郑重声明：所提交的毕业设计（论文），是本人在导师指导下，独立进行研究工作所取得的成果。除文中已注明引用的内容外，本毕业设计（论文）不包含任何其他个人或集体已经发表或撰写过的作品成果。对本研究做出过重要贡献的个人和集体，均已在文中以明确方式标明并表示了谢意。

论文作者签名：

日期：　　年　　月　　日

摘　要

随着信息网络的不断发展，可以生成大量的网络数据并执行广泛的数据分析任务。但是，用户关系网络上的数据非常敏感，必须谨慎处理以保护隐私。差分隐私是为表格数据开发的隐私标准，它提供了强大的隐私保护，而无需考虑攻击者拥有多少背景。然而，差分隐私最初是针对表数据设计的，并不能很好适用于社交网络图中。

本文首先介绍了差分隐私的研究背景和研究现状，引出了如何满足网络图结构的差分隐私处理发布的隐私与功用的平衡需求，接着介绍了差分隐私的基本定义、差分隐私的两种实现机制和两种组合性质，最后将差分隐私扩展到图领域，介绍了图的差分隐私基本定义和亟待解决的问题。

然后提出了带权值的社交网络图匿名保护方案。首先介绍了 R-MAT 生成图算法的相关知识，用于生成符合真实社交网络规律的随机网络图。然后基于约束模型设计了差分隐私扰动模型，既增加边噪音又增加节点噪音，对方案的具体细节做了详细介绍和理论分析。最后对该方案的隐私保护效用进行了分析。

之后，为了增加社交网络图匿名算法的实用性和可推广性，实现数据管理员与系统的交互性，计了一套匿名保护原型系统，采用 B/S 架构，前端采用 HTML 和 CSS 搭建，后端采用 Java 处理，通过 ajax 方法调用，涵盖了数据输入、数据加密、效果监测和数据发布 4 个功能模块，实现输入与结果的上传与下载功能。

另外，本文对匿名保护方案进方案评估，与基于密度的搜索与重建算法(DER)进行对比，分别在 4 个数据集上同时使用 DER 算法和本文提出的方案进行扰动。通过性能分析和聚集系数、边数、三角形数和平均最短路径等 4 个方面进行对比，体现了本方案在相同的隐私预算下有更好的数据可用性。

最后，本文对论文进行了总结说明，提出了本方案存在的一些不足之处，并对未来的一系列工作进行了展望。

关键词：差分隐私；权值；社交网络图；边节点扰动；原型系统

ABSTRACT

With the development of information networks, a large amount of network data can be generated and many data analysis tasks can be performed. However, the data on the user-relational network is very sensitive and must be handled with care to protect privacy. Differential privacy is a privacy standard developed for tabular data. It provides strong privacy protection without considering how much background the attacker has. However, differential privacy does not apply well to social network diagrams.

This thesis first introduces the research background and research status of differential privacy and introduces how to satisfy the balance requirements of privacy and functionality of differential privacy processing publishing in network graph structure. Then, it introduces the basic definition of differential privacy, two different implementations of differential privacy and two kinds of combinatorial properties, Finally, it extends the differential privacy to the graph area, introduces the basic definition of the differential privacy of the graph and the problems to be solved urgently.

Afterwards，an anonymous social network protection scheme with weights is proposed. Firstly, the relevant knowledge of R-MAT generation graph algorithm is introduced to generate a random network graph that conforms to the laws of real social networks. Then, based on the constraint model, the differential privacy perturbation model is designed, which not only increases the edge noise but also increases the nodal noise. The details of the scheme are introduced in detail and analyzed theoretically. Finally, the author analyzes the effectiveness of the program's privacy protection.

Next, in order to increase the utility and scalability of the anonymous algorithm for social network graphs and to realize the interaction between the data administrator and the system, an anonymous protection prototype system is adopted. The B/S architecture is adopted and the front end is constructed using HTML and CSS. The back end adopts Java processing and calls through ajax method, covering four function modules of data input, data encryption, effect monitoring and data release, and realize the function of uploading and downloading of input and result.

In addition, the anonymous protection scheme is evaluated and compared with the density-based exploration and reconstruction algorithm (DER). The DER algorithm and the proposed scheme are used to perform the perturbation on the four datasets. Through comparison of performance analysis and aggregation coefficient, number of sides, number of triangles, and average shortest path, it shows that the solution has better data availability under the same privacy budget.

Finally, this thesis summarizes the paper and puts forward some deficiencies in the program and looks forward to a series of future work.

Keywords: Differential privacy; Weights; Social network diagram; Edge node disturbance; Prototype system

目录

第一章　绪论

1.1 研究背景及意义

随着移动终端及网络技术的更迭，移动社交网络在世界范围内发展极为迅速，国内移动社交软件微信的注册用户已超 10 亿，微博的活跃用户数也日益增加至超大规模的数据量级[1]。社交网络平台为人们提供分享、交流信息等服务，同时用户的真实信息和敏感数据要被社交网络收集和归档，随之而来的是各种各样的隐私泄露问题，例如近日 Facebook 的隐私泄露事件。

然而，随着超大规模社交网络的出现，需要在大量的社交关系中划分出不同敏感程度的边关系，并为这些边关系赋予不同的权值。相比于无权值的社交网络处理方法，能有效减少需要扰动的边数量。然而，网络中的边权值也包含着许多重要的隐私信息。例如，对某个社交网络群体进行传染病或者遗传病研究时，个体间的关系强弱可能会决定传染或者遗传的扩散趋势，这对于个体而言是极其隐私的信息。为了保护用户的隐私，社交网站经常允许用户设置信息隐私权限。例如，在 Facebook 和 Twitter 上，用户可以指定允许申请添加为好友的用户类型，并且可以设置浏览个人网站信息的权限。但是，这些设置的操作过程过于冗杂和复杂，不会引起用户的注意，也不能做出严格的隐私保证[2]。

针对带权值的社交网络的数据隐私保护，目前研究者们提出了多种对边权值扰动的方法，然而这些方法不能有效的解决图结构攻击的问题：当攻击者掌握某节点的邻接度序列时，仍然可以确定该节点在图中的位置，子图攻击仍然奏效[3]。

1.2 研究现状

关于社交网络中的隐私保护问题，现存的方法主要有两类：一是基于聚类的方法，如 Casas-Roma 等人提出的基于 *k*-匿名的保护方法，这一类方法主要是将节点（边）按规则分为不同组，并隐藏组内的详细信息[3]，能实现较高的隐私保护程度，但隐藏子图的内部信息严重影响了社交网络的局部结构分析，从而降低了数据可用性，因此如何进行有效的分组以提高数据的可用性成为这类方法需要解决的最大问题。另一类是基于网络图结构的扰动算法，如通过添加、删除和交换等操作修改网络图结构，使得发布数据和原始数据产生差异从而起到隐私保护作用，同时也保持了社交网络的原有规模，相比于聚类方法具有较高的数据可用性。其后，Dwork 提出差分隐私的概念，能实现数据的强隐私保护。前述的两类方法可以很好结合差分隐私的模型和定义，例如，通过添加、删除等操作修改网络结构图并使其满足差分隐私的需求。

1.3 本人所做的工作

首先针对带权社交网络图，设计出合理的隐私保护技术框架；其次，基于差分隐私机制提出了新的解决方案，满足网络图结构的差分隐私处理发布的隐私与功用的平衡需求，同时选择合理参数衡量该方案的隐私保护力度以及对数据可用性的影响；最后，通过仿真模拟的方式对该新方案进行有效性评估。

本人主要做了如下工作：

1. 查阅了大量相关文献，掌握了现有的图差分隐私方案及特征，在现有的研究理论基础上，提出了一种带权值的大规模社交网络数据隐私保护方法。

2. 根据所提出的数据隐私保护方法的特性，设计编写了一个原型系统，可视化地呈现所提出方案的可操作性及优越性。

3. 开展了具体实验对本方案进行仿真验证，从性能和数据可用性方面与现有方法进行比较，实验结果表明，本文方法具有较高的运行效率，能适用于更大规模的移动社交网络数据集。

1.4 本文的组织结构

本文共分为六章，每章的结构概述如下：

第一章，绪论。主要介绍本文的研究背景和研究现状，引出了如何满足网络图结构的差分隐私处理发布的隐私与功用的平衡需求

第二章，相关知识。本章首先介绍了差分隐私的基本定义，随后陈述了差分隐私的两种实现机制和两种组合性质，最后将差分隐私扩展到图领域，介绍了图的差分隐私基本定义和亟待解决的问题。

第三章，带权值的社交网络图匿名保护方案。本章首先介绍了 R-MAT 生成图算法的相关知识，用于生成符合真实社交网络规律的随机网络图。然后基于约束模型设计了差分隐私扰动模型，既增加边噪音又增加节点噪音，对方案的具体细节做了详细介绍和理论分析。最后对该方案的隐私保护效用进行了分析。

第四章，匿名保护原型系统设计。为了增加社交网络图匿名算法的实用性和可推广性，实现数据管理员与系统的交互性，计了一套匿名保护原型系统，采用 B/S 架构，前端采用 HTML 和 CSS 搭建，后端采用 Java 处理，通过 ajax 方法调用。详细分析了该系统的需求和实现过程。经测试，该系统实现了数据管理员与系统的交互性，涵盖了数据输入、数据加密、效果监测和数据发布 4 个功能模块实现输入与结果的上传与下载功能。

第五章，方案评估。首先介绍了基于密度的搜索与重建算法(DER)，然后分别在 4 个数据集上同时使用 DER 算法和本文提出的方案进行扰动。通过性能分析和聚集系数、边数、三角形数和平均最短路径等 4 个方面进行对比，体现了本

方案在相同的隐私预算下有更好的数据可用性。

第六章，总结和展望。本章对论文进行了总结说明，提出了本方案存在的一些不足之处，并对未来的一系列工作进行了展望。

第二章　相关知识

2.1 差分隐私基本定义

现有的基于匿名的隐私保护模型需要特殊的攻击前提和特定的背景知识，无法量化隐私保护的强度，因此在实际应用中存在很大的局限性。Dwork 等人提出了隐私保护模型，即差分隐私模型，可以有效解决上述限制。差分隐私的基本定义如下：

定义 2-1（ε-差分隐私）对于存在且仅存在一条记录相异的兄弟数据表 D_1 与 D_2，有随机算法 $M:D \to S^K$，$Range(M)$ 用来表示算法的所有可能的输出结果集，即对于 D_1 与 D_2 的算法输出结果 $S \in Range(M)$，若满足 ε-差分隐私，则有：

$$Pr[M(D_1) \in S] \leqslant e^{\varepsilon} Pr[M(D_2) \in S] \tag{2-1}$$

式(2-1)中概率 $Pr[\cdot]$ 表示隐私被揭露的风险，由算法 M 进行随机性的控制，隐私保护程度由隐私预算参数 ε 表示，ε 越小则隐私保护程度越高，表示邻近数据集结果越接近不容易区分，但却是以增加噪音为成本代价。

由于差分隐私模型通过对发布数据进行随机扰动，因此从统计意义上讲，无论是否存在背景知识，都不能识别该记录是否在原始数据表中。这种模式的优点是它不需要特殊攻击的前提，不关心攻击者的背景知识，并提供了一个定量分析，显示隐私泄露的风险。

差分隐私的严格数学定义保证了无论单条数据纪录 r 是否存在于数据表 D 中，算法 M 的输出内容的概率几乎不变，而差分隐私框架下的差分隐私系数 ε 一定程度上决定了它们的相似度。

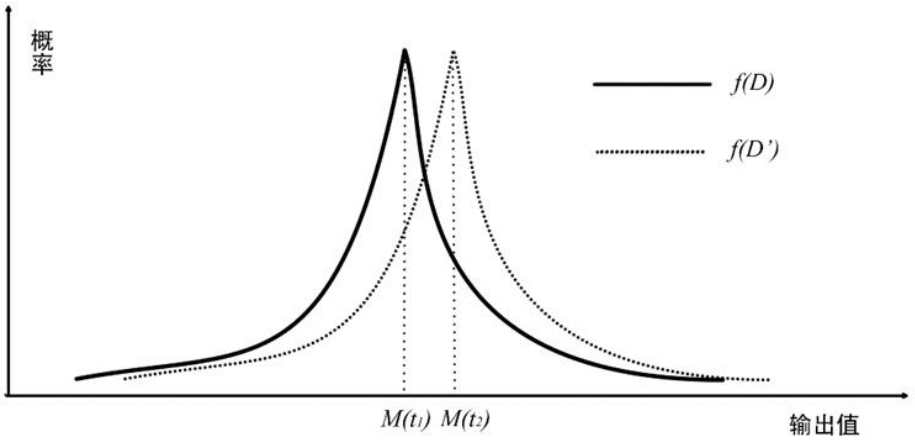


图 2.1　差分隐私随机算法输出概率密度函数示意图

图 2.1中为算法 M 对数据集 D_1 和 D_2 的输出概率密度函数示意图，其中 D_1 和 D_2 为存在且仅存在一条记录相异的兄弟数据表。在这两个数据集中，算法 M

的输出概率密度非常相似，并且攻击者仅使用返回的结果来确定数据集 D_1 和 D_2 是否不同，这是不容易的。差分隐私策略通过保护数据库中的记录免受攻击者的获知而保护用户的隐私[4]。

2.2 差分隐私的实现机制

噪声机制是实现差分隐私保护模型的常用技术。我们通过为查询功能的返回结果添加适当的噪音来实现期望的隐私保护。常用的噪声引入机制是拉普拉斯机制和指数机制，而拉普拉斯机制经常应用于数值数据。通常，根据处理对象的要求，选择适当的机制。实现差分隐私保护算法所需的噪声量与该函数的敏感度密切相关[5]。

2.2.1 Laplace 机制

拉普拉斯机制的本质是将拉普拉斯噪声 η 添加到查询函数 $f(t)$ 的返回值中，最终查询结果为 $f(t)+\eta$。其中 η 是满足拉普拉斯分布的连续随机变量，其概率密度函数是：

$$p(\eta)=\frac{1}{2\lambda}e^{-\frac{\eta}{\lambda}} \tag{2-2}$$

图 2.2是 Laplace 机制基本原理的示意图。

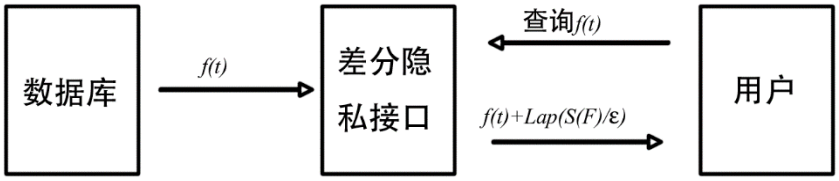


图 2.2　Laplace 机制基本原理

定义 2-2（Laplace 机制的敏感度）给定一个函数集合 F，若 $f(t)_{f\in F}\in R$，则 F 的敏感度定义为：

$$S(F)=\max_{t_1,t_2}(\sum_{f\in F}|f(t_1)-f(t_2)|) \tag{2-3}$$

t_1 和 t_2 两者相互之间最多相差一条记录，即 $|t_1\Delta t_2|\le 1$。

定理 2-1 对于函数集合 F，敏感度表示为 $S(F)$，F 的算法为向函数集合中每个 f 的输出添加独立噪音。算法 K 满足 ε-差分隐私当且仅当添加的噪音服从参数值 $S(F)/\varepsilon$ 的 Laplace 分布[6]。算法 K 满足 $S(F)/\lambda$ 差分隐私当且仅当添加的噪音服从参数 λ 的 Laplace 分布。

2.2.2 指数机制

McSherry 和 Talwar 提出了一种指数机制，它使用随机抽样方法来满足某种分布以实现差分隐私，并且可以更广泛地扩大差分隐私领域。指数机制通过定义效用评估函数q来计算每个可能结果的值。这个值的大小决定了所选概率的大小，并在保证数据发布质量的同时引起干扰。评估函数q的选择对于整个算法的噪声水平和数据的可用性起到非常重要的作用，并且通常选择为具有较低的灵敏度函数[7]。

定义 2-3 (指数机制的敏感度) 给定一个实用性评估函数q，则q的敏感度定义为

$$S(q)=\max_{t_1,t_2,r}(\left|q(t_1,r)-q(t_2,r)\right|) \tag{2-4}$$

其中t_1和t_2两者相互之间最多相差一条记录，即$\left|t_1\Delta t_2\right|\le 1$。

定理 2-2 给定数据表t,q表示数据表t所有输出实用性评估函数的集合。对数据表t和函数q来说，如果算法K满足输出为r的概率与$\exp(\frac{\varepsilon\cdot q(t,r)}{2\cdot S(q)})$成比例关系，那么就可以说匿名算法$K$满足$\varepsilon$-差分隐私[8][17]。

2.3 差分隐私的组合性质

差分隐私保护共有两种组合性质，即将不同的差分隐私算法相组合后表现出来的差分隐私特性，分别为序列组合性和并行组合性。

性质 2-1（序列组合性）假设有n个随机的算法$K_1,K_2,\cdots,K_n$，分别满足隐私预算为$\varepsilon_1,\varepsilon_2,\cdots,\varepsilon_n$的差分隐私，那么对于任一数据集$D$，组合算法$\{K_i\}\,(1\leqslant i\leqslant n)$满足$\sum_{i=1}^{n}\varepsilon_i$-差分隐私保护，即组合算法序列的隐私预算为组成序列的全部预算之和[9]。

性质 2-2（并行组合性）假设有n个随机的算法$K_1,K_2,\cdots,K_n$，分别满足隐私预算为$\varepsilon_1,\varepsilon_2,\cdots,\varepsilon_n$的差分隐私，那么对于一组数据集$D_1,D_2,\cdots,D_n$（任意两个数据集不相交），则组合算法$\{K_i\}\,(1\leqslant i\leqslant n)$满足$\max_{1\le i\le n}\{\varepsilon_i\}$-差分隐私保护，即组合算法序列的隐私预算为组成序列的最大隐私预算[9]。

2.4 图的差分隐私保护

在现今的研究中，人们一般采用两种常用的标准对图结构进行差分隐私保护，

分别是边差分隐私和节点差分隐私。

2.4.1 边差分隐私保护

对所有图$G_1=(V_1,E_1)$，图$G_2=(V_2,E_2)$，V_1,V_2,E_1,E_2分别表示图G_1和G_2的顶点集合和边集合，其中有$V_2=V_1,E_2=E_1-E_x,|E_x|=k$；若图的查询函数$Q$满足差分隐私保护，则称$Q$满足$k$-边差分隐私保护[10]。

根据边差分隐私的定义，可以得知，如果已知社交网络G，则其邻近图G'即为从G中任意添加或删除k条边得到的结果[11]。由此，个体节点x和y之间是否有关系便能得到保证。

2.4.2 节点差分隐私保护

对所有图$G_1=(V_1,E_1)$，图$G_2=(V_2,E_2)$，V_1,V_2,E_1,E_2分别表示图G_1和G_2的顶点集合和边集合，其中有$V_2=V_1-x,E_2=E_1-\{(v_1,v_2)\,|v_1=x\vee v_2=x\}$，$x\in V_1$；若图的查询函数$Q$满足差分隐私保护，则称$Q$满足节点差分隐私保护[12]。

根据节点差分隐私的定义，可以得知，如果已知社交网络G，则其邻近图G'即为从G中任意添加或删除一个节点和连接该节点的所有边得到的结果[13]。由此，个体节点x是否出现在图中便不能确定。

第三章　带权值的社交网络图匿名保护方案

差分隐私保护的定义是建立在传统数据库上的，要保证查询结果不会因为数据集中某条数据的添加或删除而被影响。社交网络中的社会关系模型一般采用节点和边来表示，社交网络中的用户个体用节点来表示，用户个体间的活动或关系一般用边来记录，而用户之间活动的多少或关系的亲疏则对应着权值的大小。

3.1 R-MAT 生成图算法

近年来，人们发现，万维网、互联网拓扑以及 Peer-to-Peer 网络遵循着一定的规律，表现出“蝴蝶结”或“水母”结构，同时具有小直径特征。找到并总结出真实图所满足的规律，用几个参数捕捉每个网络图的本质，便有可能通过模型算法快速生成逼真的网络图[13]。

由 Erdös 和 Rényi 提出的生成图算法最为著名，但它不满足上述规律。现存的生成图算法可以分为两类：基于度的和基于过程的[14]。然而基于度的生成图算法仅仅考虑图的结构，并没有考虑到网络图的其他特征（如小直径，特征值等）。另一方面，基于过程的生成图算法试图用一种简单的机制来生成具有真实图的属性的网络图，R-MAT 算法即为其中一种。生成图算法得到的图应该满足：

1. 符合度的分布规律；
2. 呈现出“社区”结构；
3. 直径较小，并与其他标准相符。

N 个节点图的邻接矩阵 A 是 $N*N$ 的矩阵，如果边 $(i,\ j)$ 存在，则 $a(i,\ j)=1$，否则为 0。表 3.1 说明了 R-MAT 算法中使用的符号。

表 3.1　R-MAT 算法符号说明

符号	含义
N	实际图中的节点个数
2^n	R-MAT 图中的节点个数
E	实际图中的边数，即去掉重复边后 R-MAT 生成图中的边数
(a,b,c,d)	在 R-MAT 模型中边缘落入分区的概率，$a+b+c+d=1$

R-MAT 算法的基本思想是递归地将邻接矩阵细分成四个相等大小的分区，并以不等的概率将边分布在这些分区内：从一个空的邻接矩阵开始，我们将边逐个“丢入”矩阵内。每条边分别以 a，b，c，d 概率投入四个分区中的某一个（见图 3.1）[15]。当然，$a+b+c+d=1$。选择的分区再次被细分为四个更小的分区，重复该过程直到到达一个最小的单元格（1×1 分区）。这就是边占用的邻接矩阵

的单元格。

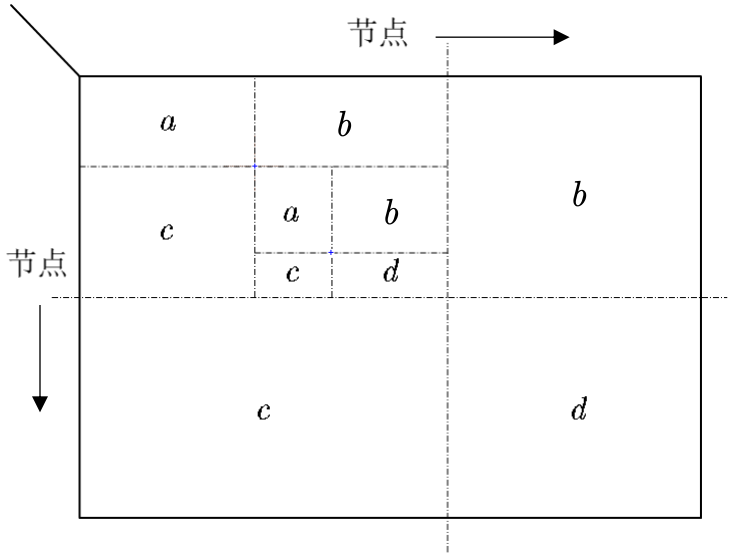


图 3.1　R-MAT 生成图算法示意图

R-MAT 算法图中的节点数设置为2^n，通常$n=\lceil \log_2 N \rceil$。有一点值得注意，可能有重复的边，即落入邻接矩阵中同一个单元的边，但我们只保留其中的一个。为了消除度分布中的波动，我们在递归的每个阶段向a，b，c，d的值添加一些噪声，然后重新归一化（以便$a+b+c+d=1$）。

通常令$a \geq b$，$a \geq c$，$a \geq d$。直观上，R-MAT 算法可以实现在图中生成“社区”结构。

1. 分区a和d代表与社区相对应的单独的节点组，比如足球爱好者和汽车爱好者。

2. 分区b和c是这两组之间的交叉部分，那里的边表示兴趣不同的朋友。

3. 分区的递归性意味着我们可以自动获得现有社区内的子社区，比如汽车爱好者内的摩托车爱好者和小汽车爱好者。

无向图必须具有对称的邻接矩阵。可以通过生成一个满足$b=c$的有向图，然后对得到的邻接矩阵的一半进行翻转，也就是复制粘贴主对角线下方的部分并丢弃上方的一半矩阵。由于$b=c$，最终的无向邻接矩阵中的边数约等于有向图中的边的数量，这也保证了结果矩阵将是对称的，因此相应的图将是无向的。至于加权图，只需要设置权重等于重复边数便可以实现[16]。

3.2 差分隐私扰动模型

社交网络分析是指利用统计方法、图论等技术对社交网络服务中产生的数据进行定量分析，通过分析个人的网络地图可以挖掘出人们在联络、信息流动与价

值交换等互动过程中潜藏的价值[17]。

3.2.1 约束模型

假设在一个带有边权值的社交网络图中，图的一系列属性可以用边权值的线性组合来表示，则当我们改变边权值且保证它仍然满足原来的线性关系时，图的属性并不会被改变。基于这一假设，Das 等人提出可以通过建立线性不等式模型来反映图属性[18]。

给定原加权图$G=(V,E,W)$，其中E为图中全体边的集合，V为全体节点集合，W为边对应的权值集合。模型的目标是利用边权值间的关系来模拟线性不等式系统。例如，在构建最小代价生成树的 Kruskal 算法中，每次选择剩余边集合中权值最小的边，同时保证不产生回路，将其添加到生成树中[19]。令(u,v)为在第i次迭代中选择的边，(u',v')为第$(i+1)$次迭代中选择的边，则有$w(u,v)\leqslant w(u',v')$。设$x_{(u,v)}$和$x_{(u',v')}$为模型中代表边的变量，则上述过程可建立不等式模型

$$x_{(u,v)}\leqslant x_{(u',v')} \tag{3-1}$$

因此，对于连续迭代过程中每次选择的边对(u,v)和(u',v')，只要给定权重满足$w(u,v)\leqslant w(u',v')$，都可以添加不等式$x_{(u,v)}\leqslant x_{(u',v')}$到模型中。

显然其他约束条件也可转化成该形式添加到模型中[20]，如公式(3-2)所示记作$AX\leqslant B$，最终构建的模型可以通过其中一组或多组约束不等式体现出原图的属性。

$$\underbrace{\begin{pmatrix} a_{11} & a_{12} & \cdots & a_{1m} \\ a_{21} & a_{22} & \cdots & a_{2m} \\ \vdots & \vdots & \ddots & \vdots \\ a_{k1} & a_{k1} & \cdots & a_{km} \end{pmatrix}}_{\mathbb{A}} \underbrace{\begin{pmatrix} x_1 \\ x_2 \\ \vdots \\ x_m \end{pmatrix}}_{\mathbb{X}} \leqslant \underbrace{\begin{pmatrix} b_1 \\ b_2 \\ \vdots \\ b_m \end{pmatrix}}_{\mathbb{B}} \tag{3-2}$$

如果边权值被重新分配为(3-2)中的不等式系统的任何解，则将确保被建模的算法的图属性保持不变。因此，该模型可以表示为线性规划问题：

$$\begin{aligned} &Max\ F(x_1,x_2,\cdots,x_m) \\ &s.t.\ AX\leqslant B \end{aligned} \tag{3-3}$$

其中 F 是线性目标函数对应于图的属性。因此图中任何能表示为边权值的线性组合的属性问题都可以转化为符合该模型的线性规划问题。而目前社交网络分析中使用的属性参数都可以用边权值的线性组合表示[21]。如果通过匿名化，模型的属性被保留，该模型被认为是正确的，即模型的任何解都可以确保被建模的属性在原图中与在匿名图中一样。因此这种模型可以完美应用于社交网络分析领域，

并且在模型建立之后，有大量完善的线性规划问题求解方法来求得上述问题的解。

通过这一模型可以轻易的解决传统隐私保护方式会改变图属性的问题，只需保证数据扰动之后仍然满足该模型约束即可[22]。模型的复杂度是确定模型所必需的不等式的数量即矩阵 A 的规模。矩阵 A 的列对应于系统中的变量，即图中边的数量，行对应于模型产生的不等式，显然当不等式数量大于边数时即可认为图属性被保留。

3.2.2 单源最短路径约束模型

在本节中，通过单源最短路径算法建立图的约束模型。设G为原始社交网络图，其中E为图中全体边的集合，V为全体节点集合，W为边对应的权值集合。初始化$V_0 = \{v_0\}$，其中v_0为给定的源点。通过改进的 Dijkstra 算法在每一步选择节点添加到生成树的过程中构造约束不等式[23]，具体算法如表 3.2 所示：

表 3.2　单源最短路径约束模型算法

算法 1 约束模型建立
输入：$G, E, V, W, V_0=\{v_0\}$
输出：约束集合 A 生成树序列 $T=\{t_1, t_2, \dots, t_i, \dots\}$

1. FOR $v \in V-V_0$ && $(v_0, v) \in E$
2. $Q=Q+\{v\}$
//设置 Q 集合为下一步可到达的所有点集合
3. END FOR
4. IF $Q = \emptyset$ && $V - V_0 \neq \emptyset$
//当图 G 不联通时，遍历完一个联通区域后重新设置源点
5. $V=V-V_0$
6. v_0=rand(V)
7. $V_0=\{v_0\}$
8. $Q=Q+\{v\}$
9. WHILE $Q \neq \emptyset$
10. u=minweight(Q)
//求 Q 集合中到 V_0 边权值最小的点
11. $A=A+\{f(v_0, pre\_u)<f(v_0, u)\}$
//添加约束集合，pre_u 记录上一步选择的节点，f 为查询函数，$f(a,b)$表示节点 a 到节点 b 的路径长度。
12. $pre\_u=u$
13. $V_0=V_0+\{u\}$
14. FOR $v \in V-V_0$ && $(u,v) \in E$
15. IF $D(v_0, v)>D(v_0, u)+w(u, v)$
//D 表示当前节点间的距离
16. $A=A+\{f(v_0,v)>f(v_0, u)+w(u, v)\}$
17. $D(v_0,v)=D(v_0, u)+w(u, v)$

```
18.        ELSE
19.            A=A+{f(v0,v)≤f(v0,u)+w(u,v)}
20.        END IF
21.        Q=Q+{v}
22.      END FOR
23.    END WHILE
24.    build T
25.  END IF
```

算法 1 的过程类似于 Dijkstra 算法。由于社交网络图的稀疏性，图 G 往往是非连通的，因此当算法执行到 Q 集合为空即从源点出发，剩余节点均不可到达时终止此次循环，将生成树 t_i 添加到生成树序列 T 中，重新从未添加的节点集合中选择源点，重复上述过程直到所有节点均被添加到序列 T 中。每次构建生成树 t_i 的过程与 Dijkstra 算法相同，从集合 Q 中选择出到集合 V_0 权值最小的节点记作 u，将节点 u 添加到集合 V_0 中并更新 V_0 到 Q 集合中节点的路径，根据更新的路径生成约束条件，同时 pre_u 为上一步从 Q 选择出添加到 V_0 的节点，生成约束条件 $A(u,pre\_u)$，最后使得 $pre\_u = u$，更新集合 Q[24]。其中生成约束不等式的规则如下：

1. Dijkstra 算法的执行过程是贪心选择的过程，因此 pre_u 节点作为上次步骤中选择的节点，$D(v_0,pre\_u) < D(v_0,u)$ 将必然成立。所构建的约束不等式为：

$$f(v_0,pre\_u) < f(v_0,u) \tag{3-4}$$

2. 若通过节点 u 可以优化 $D(v_0,v)$ 即 $D(v_0,v) > D(v_0,u) + w(u,v)$，则将 $D(v_0,v)$ 更新为 $D(v_0,u) + w(u,v)$，同时构建约束不等式如下：

$$f(v_0,v) > f(v_0,u) + x_{(u,v)} \tag{3-5}$$

3. 若通过节点 u 不可优化 $D(v_0,v)$ 即 $D(v_0,v) \leqslant D(v_0,u) + w(u,v)$，则构建的约束不等式如下：

$$f(v_0,v) \leqslant f(v_0,u) + x_{(u,v)} \tag{3-6}$$

已知 Dijkstra 算法优化后的复杂度为 $O((m+n)\log n)$，其中 m 表示边数，n 表示节点数，本算法对 Dijkstra 算法的改进并没有增加它的复杂度，同为 $O((m+n)\log n)$。

生成的约束不等式分为两个部分，规则 1 生成的约束不等式为第一部分，规则 2 和规则 3 生成的约束不等式为第二部分。第一部分的数量计算很简单，每次选择节点时都会生成一个约束不等式，因此第一部分的不等式个数可看作节点数，记为 $O(n)$。第二部分的约束不等式是在每次选择节点之后生成的，设 N_i 为第 i

次迭代时生成的不等式数，n_Q表示 Q 中的节点数，则$N_i = n_Q$。Q 中的节点数一直在改变，因而无法确定其具体数值，但是我们可以计算出它的上界。若已知图中节点度的平均值为 *degree*，V_0集合中的节点个数为n_V，当V_0集合中边数最少时即边数等于$n_V - 1$ (连通图边数最少为节点数-1)，Q 中节点数取得最大值，则 Q 中最大的节点数n_Q为$n_V \times (degree - 1) + 1$。令$N = \sum_{i=1}^{n} N_i$，带入$n_Q$可得 $N = \sum_{i=1}^{n} n_{V_{0_i}} \times (degree - 1) + 1$。$V_0$每次迭代的值为(1, 2, 3, …, *n*)，因此上式可化简为$N = \frac{n(n+1)}{2} \times (degree - 1) + n$。显然社交网络中图的平均度 *degree* 要远小于节点数 *N*，因此可将 *degree* 看作常数项[25]，则第一部分的约束不等式个数可记为$O(n^2)$。

3.2.3 差分隐私边噪音添加

在带权值的社交网络图中的噪声添加不同于无权值图，一条边的权值改动将会影响整个图的结构，如最短路和最小代价生成树，都会发生相应的改变。因此本文采用 3.1 节的约束模型对添加噪音进行线性约束。

在扰动过程中，由 Laplace 机制产生噪声，针对单元最短路径模型，查询函数$f(u,v)$的返回值是节点u和节点v间的最短路径，f的敏感度为：

$$S(f) = \sum_{(u',v') \in P[u,v]} w'_{(u',v')} - w_{(u',v')} \tag{3-7}$$

式中$P(u,v)$表示节点u和节点v间的最短路径的边集合，w'和w分别表示扰动前后边(u',v')和(u,v)的权值[26]。

具体添加算法如表 3.3：

表 3.3　差分隐私边噪音添加模型算法

算法 2 边噪音添加
输入：*G, A, T, ε1*
输出：*G'*
1. FOR (*u*, *v*) ∈ E_T 2. *w'*(*u*, *v*) =*w*(*u*, *v*)+Laplace(*ε*1) 3. END FOR 4. FOR (*u*, *v*) ∈ E_N 5. *w'*(*u*, *v*)=value->answer(*A*)+Laplace(*ε*1) //value->answer(*A*)为式(4)中线性规划的求解结果 6. END FOR 7. RAND$(u,v) \cap E_T = \emptyset$

//随机选择几条不存在边的节点对u, v 8. $E=E+\{(u, v)\}$ 9. $w(u, v)=\min(t_i, \max w_t)$

算法 2 的输入是需要添加噪音扰动的原始图数据G，E，V，W 序列T，以及约束集合A和隐私预算ε_1，最终得到扰动图G'。

添加边噪音的步骤如下：

1. 将边集合E分成E_T和E_N两个部分，其中E_T为构成最短生成树的边集合，E_N为剩余边组成的集合；
2. 利用算法 1 生成边权值约束不等式；
3. 对E_T中的边权值添加 Laplace 噪声，再将加噪之后的权值带入约束不等式中，可解得E_N中的边权值约束，并生成新的权值；
4. 在每个生成树t_i中，选择若干原来不存在边的节点对(u,v)构造一条新边，比较t_i中最大边权值与$f(u,v)$，选择其中较小的一个作为边权值。

3.2.4 差分隐私节点噪音添加

由于节点的添加或删除具有很大的敏感性，添加或者删除某个节点时，同时连接在该节点的边关系会随之改变，显然会引入大量的噪音。本文提出的虚假节点添加方法可以有效的降低对数据可用性的影响。对于节点u'的删除操作，函数$f(u,v)$的敏感度定义为：

$$S(f)=\sum_{u,v\in V} D'[u,v]-D[u,v] \tag{3-8}$$

公式(3-8)中$D'(u,v)$，$D(u,v)$表示删除节点后的最短路径长度和原最短路径长度，由此可见直接删除节点会造成巨大的影响。节点添加时只要保证边权值足够大即可极大程度的降低对数据可用性的影响，但是过大的边权值会很容易被攻击者识别出[27]。

本文的节点扰动方式分为如下两步：一、删除度低于阈值的节点，同时对于连接到该节点的所有节点，如果他们之间存在边关系，则修改其边权值；二、添加虚假节点，虚假节点的度等于定义的阈值。其中阈值由用户根据应用需求定义。具体算法流程如表 3.4 所示：

表 3.4　差分隐私节点噪音添加模型算法

算法 3 节点噪音添加
输入：*G, E, V, W, degree_value,* ε_2, ε_3
输出：*G'*

1. $N_{noise} = |Laplace(1/\varepsilon_2)|$
   //通过用户定义的隐私预算计算扰动节点个数
2. //删除
3. RAND(v)∈V && $v.degree$< $degree\_value$
   //随机选择度小于阈值 $degree\_value$ 的节点
4. N_{noise}= N_{noise}-1
5. FOR each node u_1, u_2∈V && (u_1, v)∈E && (u_2, v)∈E
6. IF f(u_1, u_2)>w(u_1, v)+w(u_2, v)
7. IF (u_1, u_2) ∩ $E = \emptyset$ THEN E=E+(u_1, u_2)
8. w(u_1, u_2)= w(u_1, v)+w(u_2, v)+Laplace($S(f)/\varepsilon_3$)
9. END IF
10. END FOR
11. delete (v)
    //删除节点 v，以及所有连接 v 的边
12. //插入
13. RAND(v)∈V && $v.degree$< $degree\_value$
    //随机选择度小于阈值 $degree\_value$ 的节点
14. N_{noise}= N_{noise}-1
15. new v_1
    //新建一个虚假节点
16. V=V+{v_1}
17. E=E+{(v_1, v)}
18. w(v_1, v)=average(v)
    //节点 v 的所有边权值的平均值
19. FOR each u∈V && (u, v)∈E
20. E=E+{(v_1, u)}
21. w(v_1, u)=w(u, v)+ Laplace($S(f)/\varepsilon_3$)
22. END FOR

算法 3 的输入包括了原始图数据G，E，V，W 以及由用户自己定义的节点度的阈值$degree_value$ 和隐私预算 ε_2, ε_3，输出为扰动图G'。首先执行删除扰动方式，由于查询函数对节点增删的敏感度很高，因此我们选择度小于 $degree\_value$ 的节点以减小对查询函数的影响。

添加节点噪音的步骤如表 3.5 所示：

表 3.5　添加节点噪音步骤

删除节点	插入节点
1.以度小于$degree_value$的节点为基准，选择扰动节点个数$N_{noise}=\lvert Laplace(1/\varepsilon_2)\rvert$（增加或删除节点的敏感度为 1）。	
2. 随机选择 N 个节点分别作为基准节点。	
对每一个基准节点v，对所有与其相连的节点u_1和u_2进行如下操作：	对每一个基准节点v，对所有与其相连的节点u进行如下操作：
3. 若u_1和u_2都有到v的边，且$f(u_1,u_2)=w(u_1,v)+w(v,u_2)$，则令$w(u_1,u_2)=w(u_1,v)+w(v,u_2)$；若$u_1$和$u_2$之间不存在边则构造一条边，权值为$w(u_1,v)+w(v,u_2)$。	3. 添加虚假节点v_1，并且构造边(v_1,v)，权值设置为v的边权值的平均值。
4. 删除节点v以及所有与其相连的边。	4. 将所有与v连接的节点u都构造一条边连接到节点v_1，边(v_1,u)的边权值等于$w(u,v)+Laplace(S(f)/\varepsilon_3)$。

由该算法所生成的边节点扰动对图的平均最短路径几乎没有影响，但是会增加三角计数，具体计算公式如下：

$$C_{tri}{}'=C_{tri}+\sum_{v\in V_{Fake}}(degree-1) \tag{3-9}$$

公式(3-9)中C_{tri}表示原图中的三角计数，V_{Fake}为添加的虚假节点集合。由公式(3-9)可以看出，当添加的虚假节点度越高，三角计数越高。而虚假节点的度又与构建节点时的基准节点有关，因此我们选择度较小的节点作为基准节点。

3.2.5 隐私保护分析

差分隐私算法的隐私保护程度与添加噪声的过程有关，分为两个步骤：

1. 通过约束模型，对边权值添加差分隐私噪声，隐私预算为 ε_1；

2. 节点扰动，即节点的删除和虚假节点的添加，隐私预算为 ε_2，构造噪音边权值的隐私预算为 ε_3。

首先，在算法 2 中对组成生成树的边权值全部添加隐私预算为 ε_1 的差分隐私，很显然再通过约束条件得出的权值都满足 ε_1-差分隐私。其次，在算法 3 中，节点扰动数量由 Laplace 机制计算得出，是 ε_2-差分隐私的直接应用；同理，由 Laplace 机制产生的噪音边权值也满足 ε_3-差分隐私。最后对算法 2 与算法 3 合并后的结果进行分析，由定理 2 可知，当算法 2 与算法 3 同时作用在图 G 时满足

$(\varepsilon_1+\varepsilon_2+\varepsilon_3)$-差分隐私，令$\varepsilon=\varepsilon_1+\varepsilon_2+\varepsilon_3$，则本算法符合 ε-差分隐私。

第四章 匿名保护原型系统设计

为了增加社交网络图匿名算法的可用性，提高其实用性和可推广性，实现数据管理员与系统的交互性，并直观展示，本人设计了一套匿名保护原型系统。随着互联网技术的兴起，B/S 架构的应用范围越来越广。用户只需要一个浏览器便可以享受自己的服务，而不需要下载客户端。于是，基于 Java 的匿名保护原型系统应运而生。

4.1 需求分析

第三章中提到的带权值的社交网络图匿名保护方案旨在对大规模的社交网络关系进行扰动，使得重新发布的网络图可以不至泄露单个用户的隐私。但同时，社交网络反映了现实社会的各个方面，同时为研究人类社会中的各种现象提供了宝贵而丰富的资源。为了在对社交网络进行数据挖掘的同时抵御攻击者，设计匿名保护原型系统。

匿名保护原型系统的使用者主要为大型社交网络的数据拥有者，其掌握这该社交网络中的每一个用户的真实社交信息。当数据拥有者将全部的用户社交信息输入到匿名保护原型系统后，系统返回经处理过后的用户社交信息。故匿名保护原型系统应实现表 4.1 所示基本功能。

表 4.1　匿名保护原型系统功能模块

功能模块	功能描述
数据输入	系统使用者上传其掌握的用户间信息，上传至服务器。
数据加密	系统使用者根据其期望的保护程度输入隐私预算，作为匿名保护原型系统的控制参数。
效果监测	匿名保护原型系统呈现经处理后社交网络图属性，以供参考。
数据发布	匿名保护原型系统返回经过匿名扰动过后的社交网络关系图给系统使用者。

4.1.1 数据输入需求分析

匿名保护原型系统的输入为用户节点之间社交信息，主要体现为两者之间的通信频率，反映到图论中即为原始图G中节点对(u,v)之间的权重$w(u,v)$。在数据输入时，应允许系统使用者输入每一条边的起点、终点和边权重。表 4.2 所示为数据输入的具体需求：

表 4.2　数据输入需求分析

原始图信息	对应的系统显示	
边起点	用户 ID	粉丝 ID
边终点	关注的人的 ID	用户 ID
边权重	通信频率	通信频率

图 4.1是数据输入的示意图。如图所示，每一个五元组由关注的人 ID，用户 ID，粉丝 ID 和两个通信频率组成，每一个通信频率即代表图中一条边的权值。

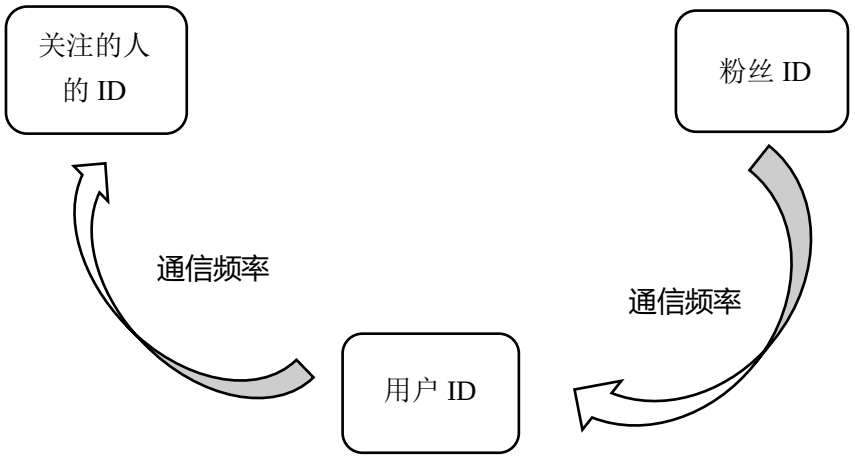


图 4.1　数据输入示意图

由于移动社交网络通常具有超大规模，如果对每一条边都手动输入起点、终点和边权值，对于系统使用者来说是一个很大的工作。因此，匿名保护原型系统的数据输入功能必须包含自动输入功能，即上传并读取文件内容，并将系统使用者所上传的文件内容转化为输入数据所需要的格式。

4.1.2 数据加密需求分析

根据第三章所述，该带权值的社交网络图匿名保护方案分别对原始图G添加了边噪音和节点噪音。算法 2 添加边噪音算法的输入是需要添加噪音扰动的原始图数据G，E，V，W序列T，以及约束集合A和隐私预算ε_1，最终得到扰动图G'。其中隐私预算ε_1为向最短生成树的边E_T添加噪音的隐私预算，需要系统使用者根据自身需要加以赋值，隐私预算ε_1越小隐私保护程度越高。算法 3 添加节点噪音的输入包括了原始图数据G，E，V，W以及由用户自己定义的节点度的阈值$degree_value$和隐私预算 ε_2, ε_3，输出为扰动图G'。其中隐私预算ε_2代表添加扰动节点的数量，ε_3代表插入节点时构造的新边的权值噪音，隐私预算ε_2越小隐私保护程度越高，隐私预算ε_3越小隐私保护程度越高。

故匿名保护原型系统数据加密模块需要系统输入的参数有 3，分别如表 4.3 所示：

表 4.3　数据加密模块输入参数

参数	名称	说明
ε_1	添加边噪音的隐私预算	隐私预算ε_1越小隐私保护程度越高
ε_2	添加删除节点的隐私预算	隐私预算ε_2越小隐私保护程度越高
ε_3	构造噪音边权值的隐私预算	隐私预算ε_3越小隐私保护程度越高

4.1.3 效果监测需求分析

效果监测模块主要用来监测该匿名保护原型系统的扰动效果，为了体现本方案的优越性，系统将本方法与基于密度的搜索与重建算法（DER）进行对比。通过性能分析和聚集系数、边数、三角形数和平均最短路径等 4 个方面进行对比，体现了本方案在相同的隐私预算下有更好的数据可用性。对比结果主要通过柱状图和折线图的方式呈现。

故匿名保护原型系统效果监测模块主要功能为输出，需要实现将对比结果图呈现到页面，具体要求如图 4.2所示。

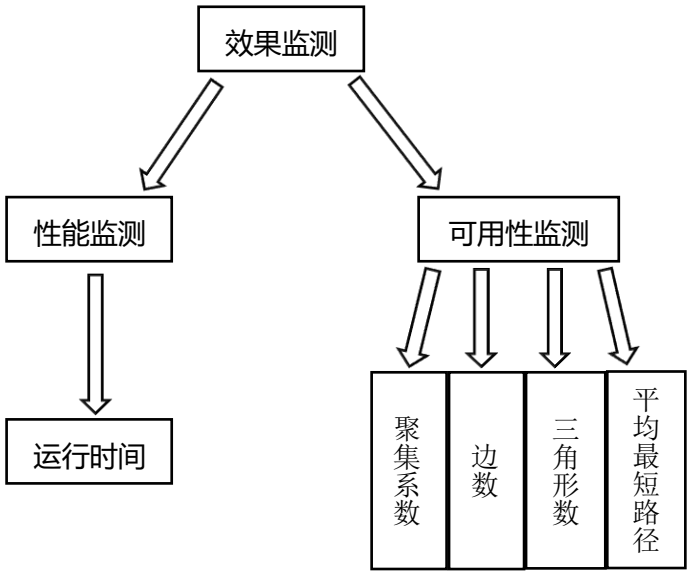


图 4.2　效果监测模块输出

4.1.4 数据发布需求分析

数据发布模块主要用于获取经匿名保护算法处理过后的社交网络图。算法的输出为一系列以三元组形式的边x_{ij}，包括边两端的节点序号i，j和边的权值w_{ij}。

由背景知识可知，社交网络图中边的含义表示社交网络中两用户i和j的个人信息相似度。为了提高系统的友好性，本系统应实现将处理过后的图G'以最直白的结果呈现给系统使用者，即逐条解释图中边的含义。

因此，数据发布的形式如表 4.4 所示：

表 4.4　数据发布形式

发布形式	图中形式
用户 A	边x_{ij}的起点i
用户 B	边x_{ij}的终点j
相似度	边x_{ij}的权值w_{ij}

与数据输入模块相似，由于移动社交网络通常具有超大规模，因此，匿名保护原型系统的数据输出支持文件下载，将处理过后的结果写入文件中，以.txt 格式保存。

由于在本方案中，采用了无向图的形式，即并未区分边的起点与终点，故在输出结果时近取邻接矩阵的上三角，对重复的输出结果仅展示一次。

4.2 系统实现

根据需求分析所示，进行匿名保护原型系统编写。其中前端页面采用 bootstrap 架构，用 HTML 语言和 CSS 实现，后端采用 Java 的 IO 方法读写文件，前后端采用 ajax 方法调用。

最终完成的系统界面如下图 4.3至图 4.11所示。

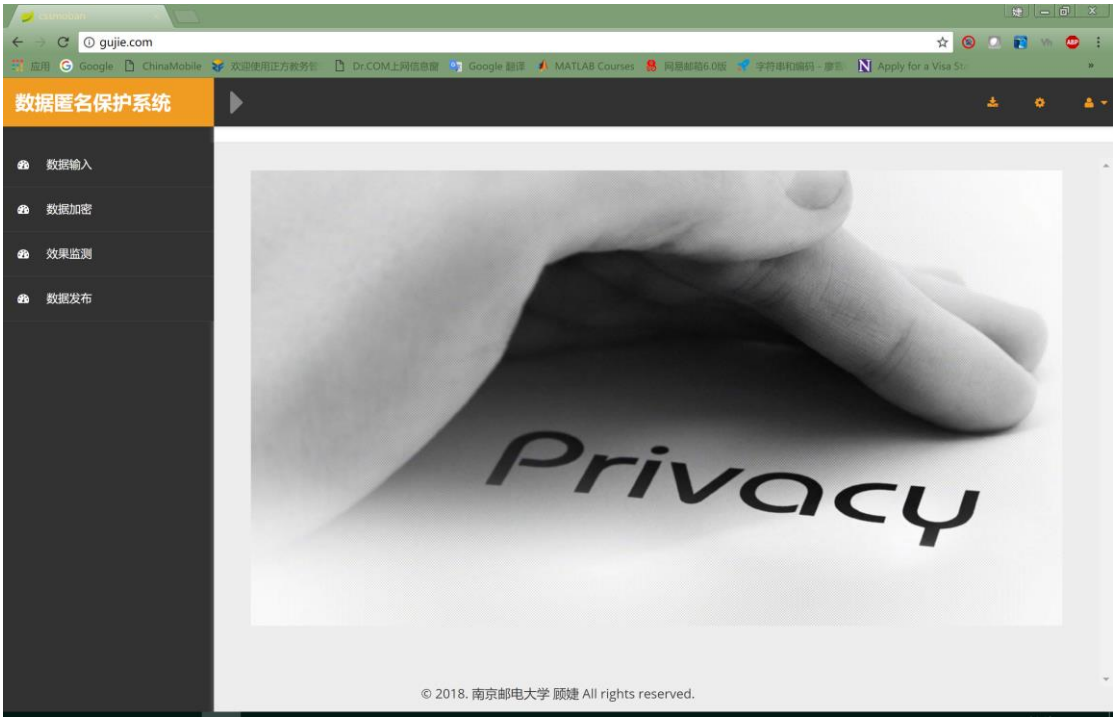


图 4.3　匿名保护原型系统首页

修改本地域名为 www.gujie.com，在浏览器中输入 www.gujie.com，进入原型系统首页，如图 4.3所示。

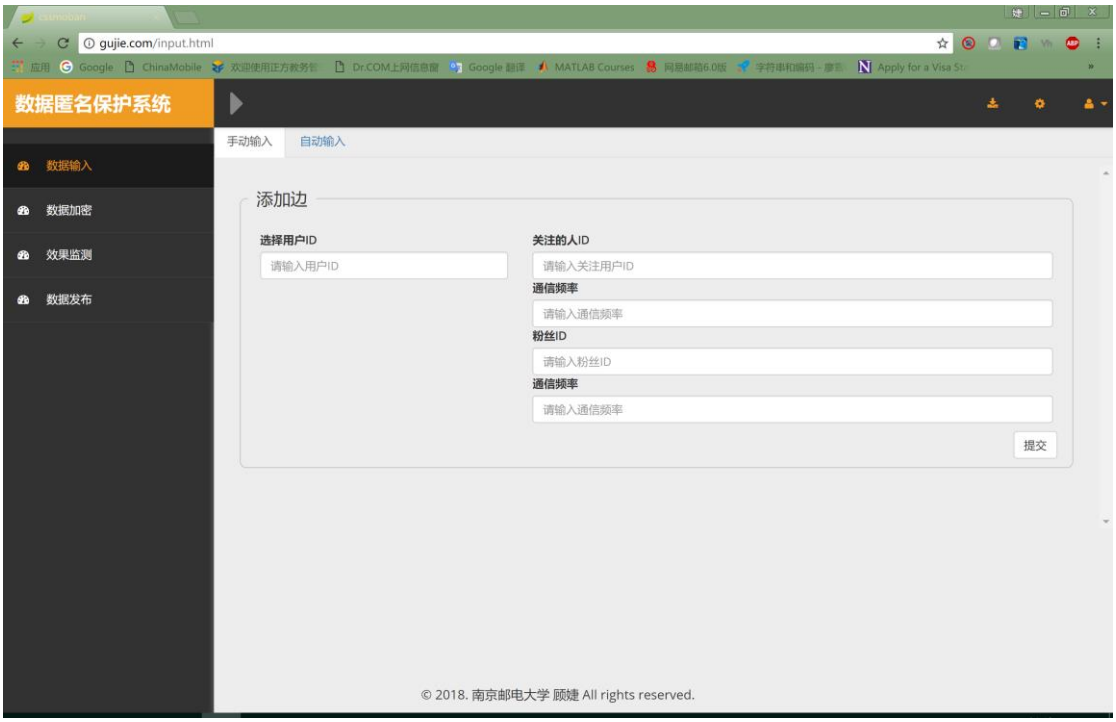


图 4.4　原型系统手动输入页面

点击左侧数据输入 tab，默认进入手动输入页面。在这个页面上，系统使用者手动输入用户 ID，关注的人的 ID，粉丝 ID 和两个通信频率。点击提交后，输入的数据通过 Java 写入本地 data.txt 文件作为数据输入，如图 4.4所示。

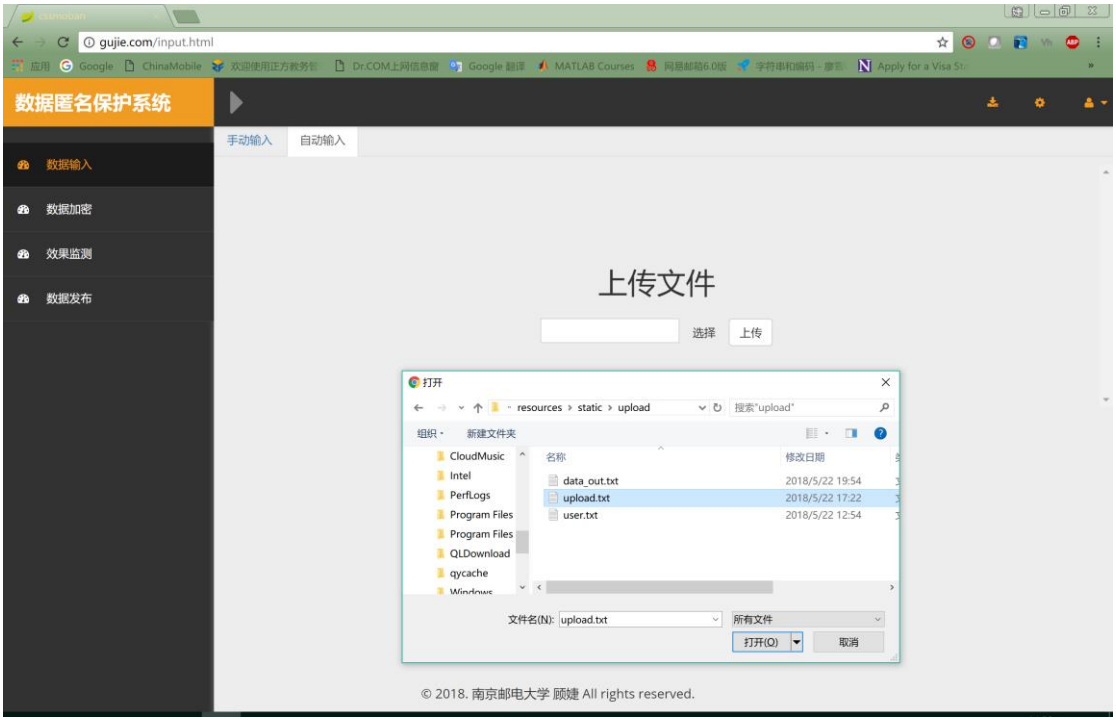


图 4.5　原型系统自动输入页面

系统使用者也可切换到自动输入 tab，通过上传本地的 upload.txt 文件将大规模的社交网络图上传至系统内部供后台 MATLAB 程序运算，如图 4.5所示。

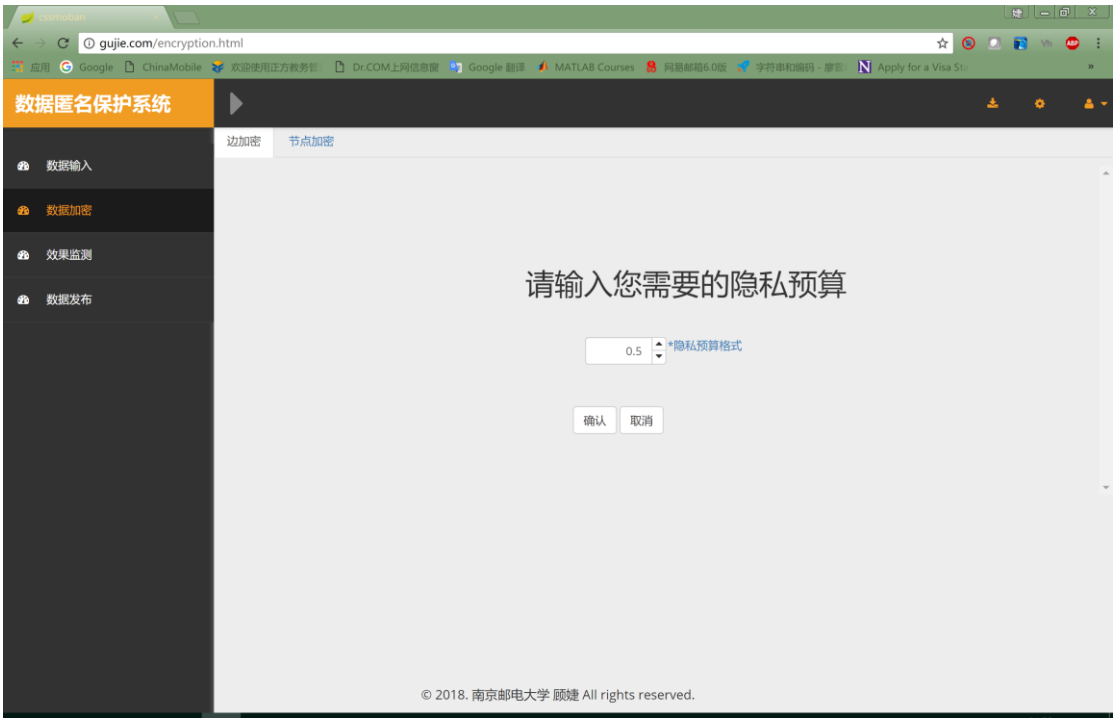


图 4.6　原型系统边加密页面

完成数据输入后，系统使用者点击左侧数据加密 tab 进入数据加密模块。首先根据隐私预算格式输入添加边噪音的隐私预算ε_1，然后切换上方节点加密 tab，输入添加删除节点的隐私预算 ε_2 和构造噪音边权值的隐私预算 ε_3。如图 4.6 和图 4.7所示。

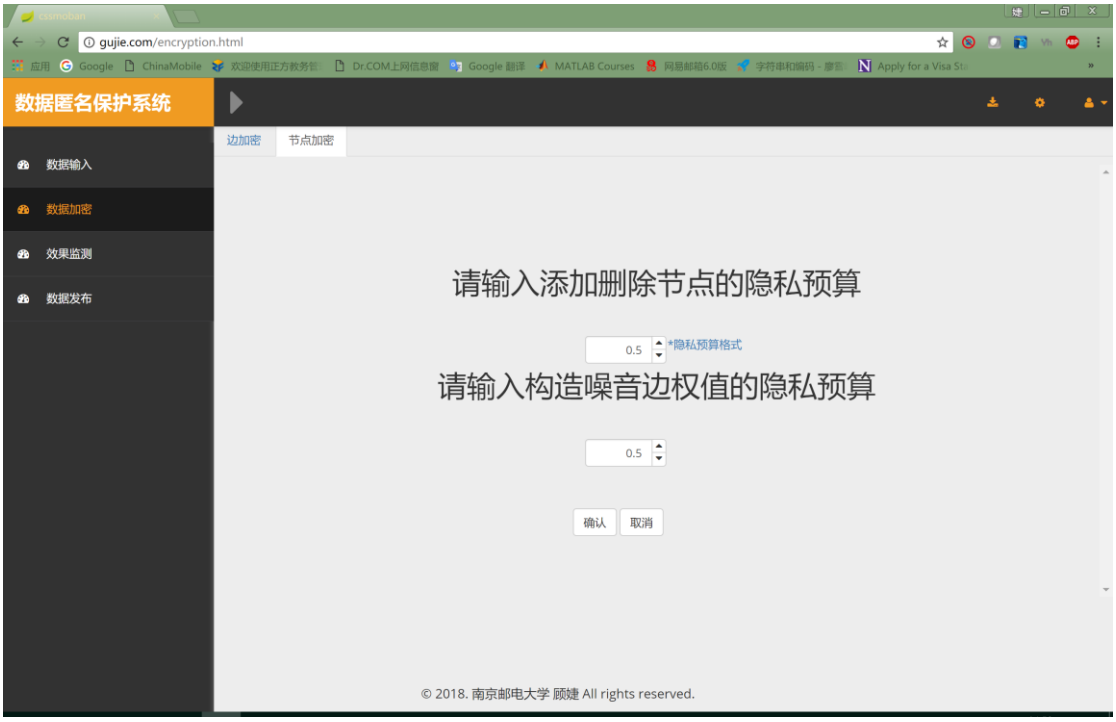


图 4.7　原型系统节点加密页面

图 4.8　原型系统性能监测显示

点击左侧效果监测 tab 进入效果监测页面，首先默认展示本文提出的带权值的社交网络图匿名保护方案和 DER 算法的运行时间对比。随后，可以通过切换上方可用性监测 tab 查看两种方案的聚集系、三角形数、平均最短路径和边数等 4 个图属性的对比。如图 4.8和图 4.9所示。

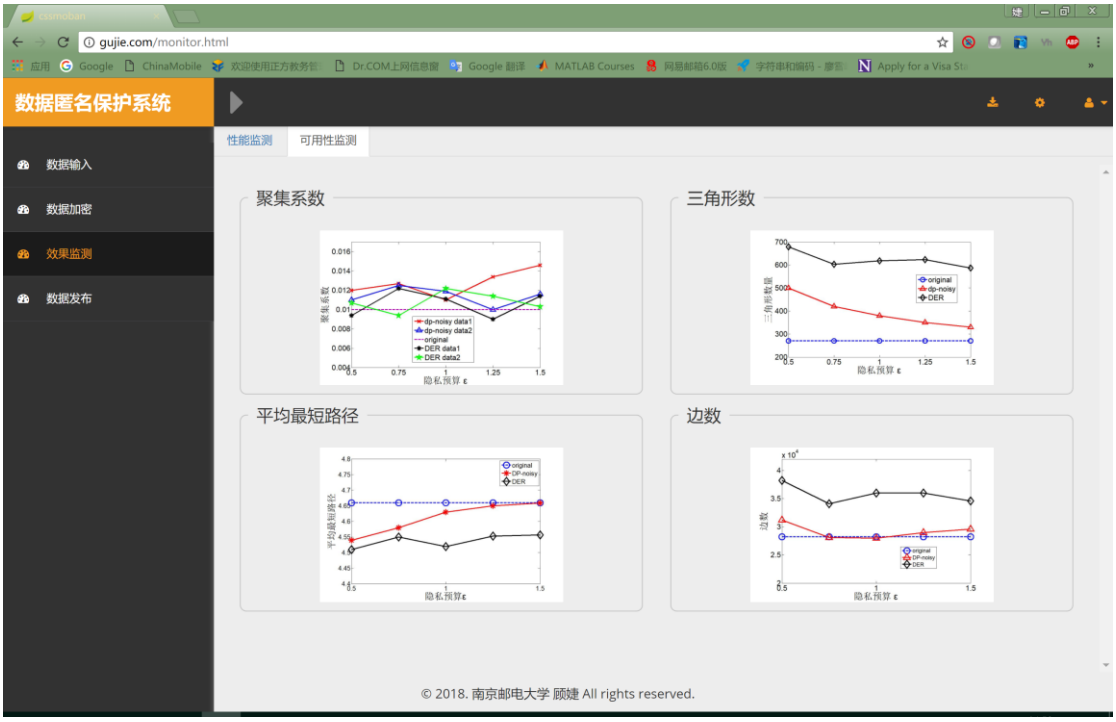


图 4.9　原型系统可用性监测显示

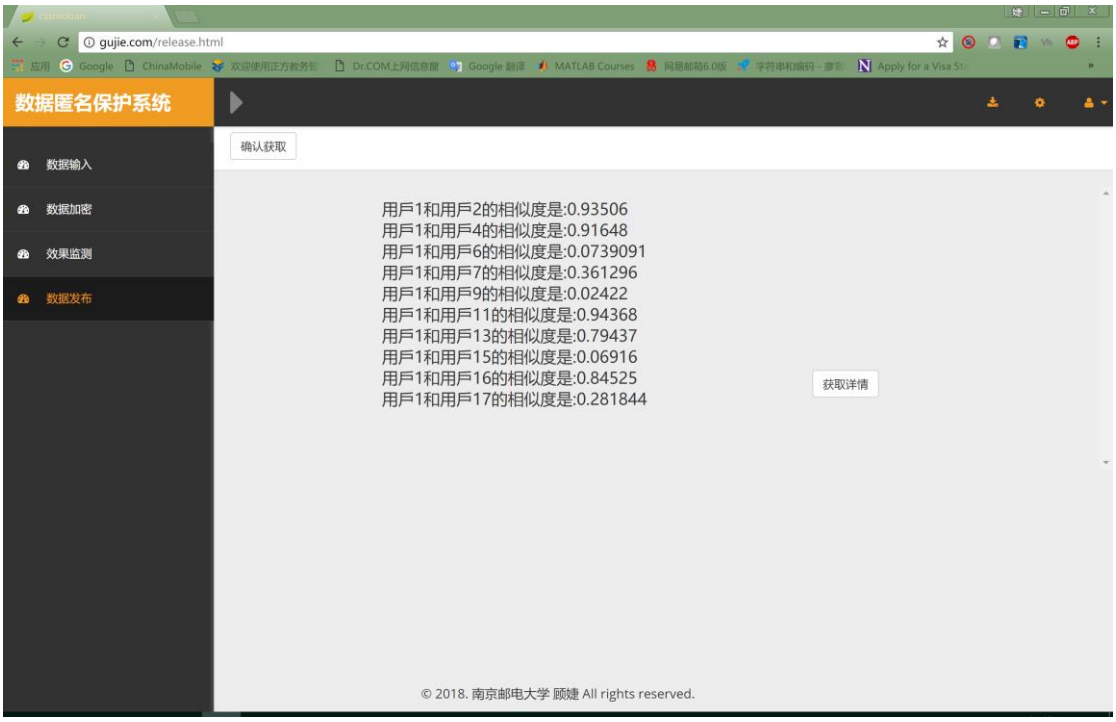


图 4.10　原型系统数据发布页面

最后，点击左侧数据发布 tab 进入数据发布页面。点击上方确认获取按键可以查看该匿名保护原型系统输出结果的前 10 条，结果采用通俗化的语言，即将系统输出的图形语言转化为用户之间的相似度结果，增强了用户友好性，如图 4.10所示。

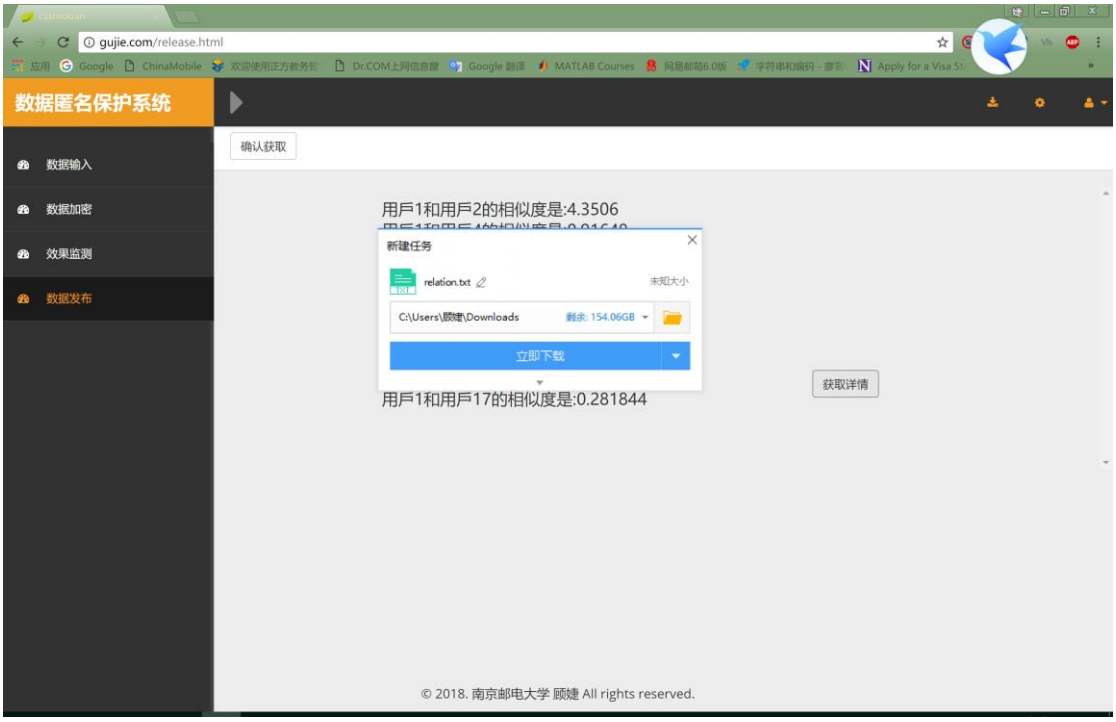


图 4.11　原型系统获取详情页面

与数据输入模块相似，由于移动社交网络通常具有超大规模，因此，匿名保护原型系统的数据输出支持文件下载，将处理过后的结果写入文件 relation.txt 中。点击右方获取详情按键，拉起迅雷下载，可以获得全部的处理输出结果，如图 4.11 所示。

经测试，该系统实现了数据管理员与系统的交互性，涵盖了数据输入、数据加密、效果监测和数据发布 4 个功能模块实现输入与结果的上传与下载功能。

第五章　方案评估

本章主要用仿真实验来分析本方法（在实验中简写为 dp-noisy）的数据可用性和执行效率。如表 5.1 所示，实验依次采用了四个用 R-MAT 生成图算法按照社交网络参数随机生成的社交图模拟数据集，节点数分别为 1000 节点，5000 节点，10000 节点和 30000 节点。隐私预算分配方式为 ε_1:ε_2:ε_3=2:1:2(ε_2 控制添加的节点噪音，因为节点数规模较大，需要添加大量节点噪音，所以 ε_2 取值分配较小)。

表 5.1　四个不同数据集的节点数和边数

数据集	节点数	边数
data1	1000	28247
data2	5000	65940
data3	10000	212781
data4	30000	932974

为了体现本方案的优越性，本文将本方法与基于密度的搜索与重建算法（DER）进行对比。

5.1 基于密度的搜索与重构算法（DER）

基于密度的搜索与重构算法通过引入一个额外的参数来衡量相关程度，使得差分隐私可以被调整为提供可证明的隐私保证，为非交互式网络数据发布提供整体解决方案。算法首先为给定的网络数据集生成一个私有顶点标签，使相应的邻接矩阵形成密集的簇。接下来，通过依赖于数据的分区过程自适应地识别邻接矩阵的密集区域。最后，通过创造性地使用差分隐私的指数机制来重建噪声邻接矩阵[28]。

表 5.2　DER 算法伪代码

算法 1 DER 算法
输入：原始图G；隐私预算ϵ；相关系数k
输出：处理后的图G'
1. $\frac{\epsilon}{k}=\epsilon_I+\epsilon_E+\epsilon_A$ 2. 顶点标签$\mathcal{L}$←确定顶点标签(G,ϵ_I) 3. 根据顶点标签$\mathcal{L}$生成基于G的邻接矩阵A 4. 噪音四叉树$\mathcal{QT}$←搜索密集区域(A,ϵ_E) 5. 处理后的矩阵A←处理边$(\mathcal{QT},A,\epsilon_A)$

6. 根据矩阵A生成图G'
7. 返回图G'

表 5.2 中是作为对比的算法的概述，称为基于密度的搜索与重构算法（DER）。它将图G，隐私预算和相关参数k作为输入，并且返回处理后的图G'，其满足基于数据库的ϵ-差异隐私并且相关性$\leqslant k$。解决方案包含三个主要步骤，将实际隐私预算调整为$\frac{\epsilon}{k}$以抵消相关性的影响，然后将$\frac{\epsilon}{k}$分为ϵ_I，ϵ_E和ϵ_A三部分，每部分用于一个步骤。

在第一步确定顶点标签中，DER 算法旨在确定一个良好的顶点标签，使相应的邻接矩阵形成 1s 的密集簇。可以达到此目的的现有方法对边的敏感度很高，因此难以实现具有可接受效用的差分隐私。于是 DER 算法采用了一种有效的贪婪算法，该算法从随机顶点标签中迭代排列顶点对以获得更好的密度对比度。

在第二步探索密集区域中，DER 算法通过调整标准四叉树来搜索图G的邻接矩阵A的密集区域，从而可以高精度地重构，从而设计一个差分私有和数据相关的分区过程。该过程导致噪声四叉树$\mathcal{QT}$，其节点表示A的区域并且与噪声计数相关联。该步骤的主要技术挑战包括基于精确估计$\mathcal{QT}$高度的停止条件的设计，基于指数机制的分裂点选择，自适应隐私预算分配方案和高效实施，其中每一点对于整个算法的成功都是关键。

在第三步处理边中，DER 算法提出了一种有效的边排列算法来重构一个噪声图形矩阵A'，该矩阵使$\sum_{i=1}^{|V|}\sum_{j=1}^{|V|}|A_{ij}-A_{ij}'|$最小[29]。该算法创造性地运用了差分隐私的指数机制，该机制为扩大输出域提供了有效的解决方案。与简单实现的阶乘复杂度相比，它成功地将时间复杂度降低到了$O(|V|^2)$。

基于密度的搜索与重构算法是第一个通过差分隐私为网络数据发布提供实用解决方案的算法。大量的实验表明，DER 算法在不同类型的实际网络数据集上执行各种数据分析任务时表现良好。

5.2 性能分析

我们分别在四个数据集上将本算法的运行时间与同样采用差分隐私保护机制的 DER 方法进行对比，实验中两种方法的隐私预算 ε 取值都为 1（其中 dp-noisy 的隐私预算分配方式为 ε_1:ε_2:ε_3=2:1:2）。图 5.1中横坐标表示数据集的大小，纵坐标表示运行时间（单位秒）。

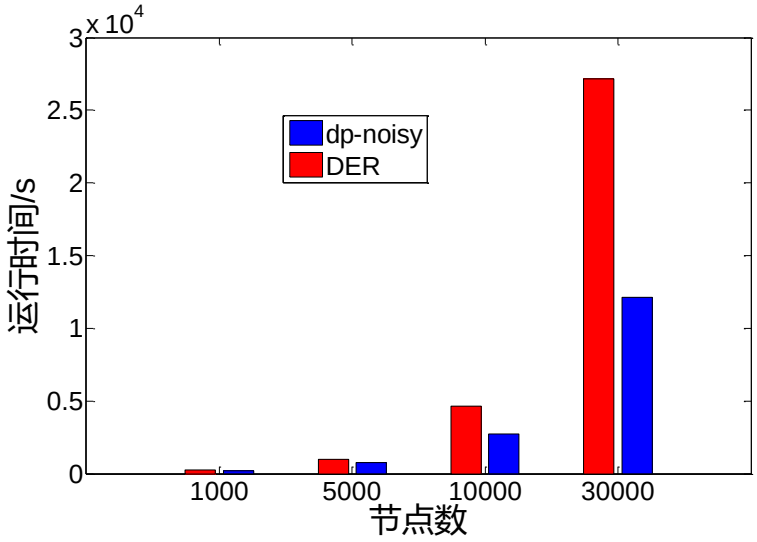


图 5.1　运行时间对比

从图 5.1可以看出随着数据集规模的增加，两种方法的运行时间都在增加，但 dp-noisy 相较于 DER 方法有了较大程度的提高，因为 DER 方法的运行时间主要是在矩阵聚集上，这部分执行的过程中，每一次交换都需要计算曼哈顿距离，时间复杂度为$O(n^2)$，而 dp-noisy 方法的时间复杂度在上文证明过为$O((m+n)\log n)$。这说明了本方法可以运行在较大规模的数据集上，且对比传统扰动方法 DER 具有良好的可扩展性。

5.3 数据可用性分析

对于带权值的社交网络图数据而言，数据可用性分析将分成以下两个部分进行：一是社交网络的结构特征参数分析，二是边权值分析。

对于结构特征参数，本实验通过将原始数据的图特征参数与扰动分布后社交网络数据的图特征参数进行比较分析，以验证在不同的隐私预算下本算法的数据可用性。本实验选择了聚集系数、三角形计数、平均最短路径和边数四个最重要的参数。为了更清晰的体现本算法对图结构的影响，此处的平均最短路径参数记录的仍然是节点数，即在确定最短路径之后，忽略边权值只记录经过的节点数。本文选择同样使用差分隐私保护机制的 DER 方法在四个数据集上进行了对比。

5.3.1 聚集系数对比

在图论中，集聚系数是图中的点倾向于集聚在一起的程度的一种度量。证据显示：在多数实际网络以及特殊的社会网络中，结点有形成团的强烈倾向，这一倾向的特征是有一个相对紧密的连接。在实际网络中，这种可能性比随机生成的

均匀网络的两个结点间连接的可能性大。

这个措施有两个版本，局部的和全局的。全局方法旨在衡量整个网络中的聚合度，而局部方法显示了单节点嵌入的度量。

全局集聚系数基于节点三元组，节点三元组是具有 2 条无向边（开放三元组）或 3 条无向边（封闭三元组）的三元组。三角形由三个封闭的三元组构成，（三角形）集中在每个节点上。全局集聚系数是所有三元组（包括开放和封闭）中封闭三元组的数量，节点的局部聚集系数表示其相邻节点组成完全图的紧密程度。

设G为原始社交网络图，其中E为图中全体边的集合，V为全体节点集合，a_{ij}表示连接结点i与结点j的边，$N_i=\{v_j:a_{ij}\in E\cap a_{ji}\in E\}$表示$v_i$的相邻结点，$k_i$表示$v_i$相邻结点的数量。

结点v_i的局部集聚系数C_i是它的相邻结点之间的连接数与它们所有可能存在连接的数量的比值。对于一个有向图，a_{ij}与a_{ji}是不同的，因而对于每个邻结点N_i在邻结点之间可能存在有$k_i(k_i-1)$条边（k_i是结点的出入度之和）。

因此，有向图的局部集聚系数为：

$$C_i=\frac{|\{a_{jk}\}|}{k_i(k_i-1)}:v_j,v_k\in N_i,e_{jk}\in E \tag{5-1}$$

无向图的为：

$$C_i=\frac{2|\{a_{jk}\}|}{k_i(k_i-1)}:v_j,v_k\in N_i,e_{jk}\in E \tag{5-2}$$

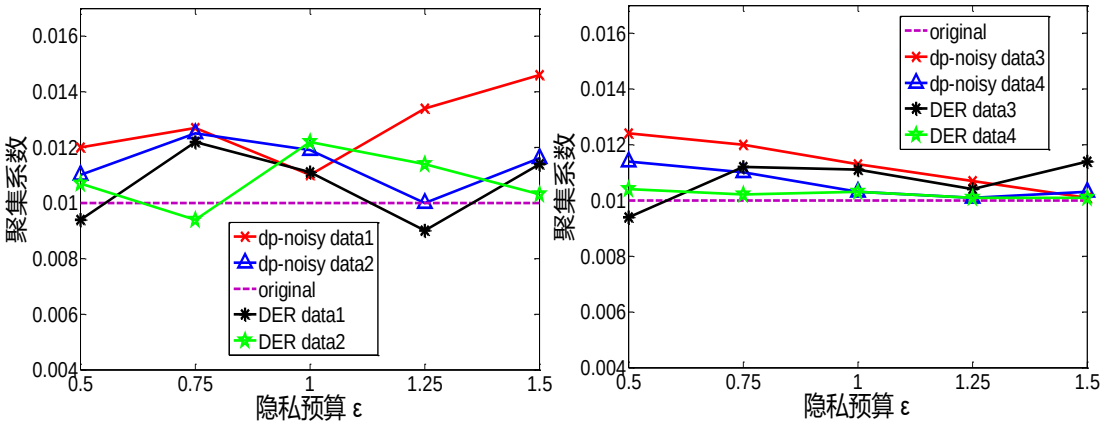
定义$\lambda_G(v)$，$v\in V(G)$为无向图G中三角形的数量，$\lambda_G(v)$是G的有三条边和三个结点的子图的数量，其中一个就是v。定义$\tau_G(v)$为$v\in V(G)$中三元组的数量。也就是说，$\tau_G(v)$是有两条边和三个结点的子图的数量，其中一个是v，这样有v两条入射边，那么可以定义集聚系数为：

$$C_i=\frac{\lambda_G(v)}{\tau_G(v)} \tag{5-3}$$

整个网络的集聚系数定义为所有结点n的局部集聚系数的均值：

$$\overline{C}=\frac{1}{n}\sum_{i=1}^{n}C_i \tag{5-4}$$

图 5.2的(a)和(b)分别是数据集 1，2 和 3，4 在 dp-noisy 算法下和 DER 算法下聚集系数的变化对比。



(a) data1 和 data2 的聚集系数　　(b) data3 和 data4 的聚集系数

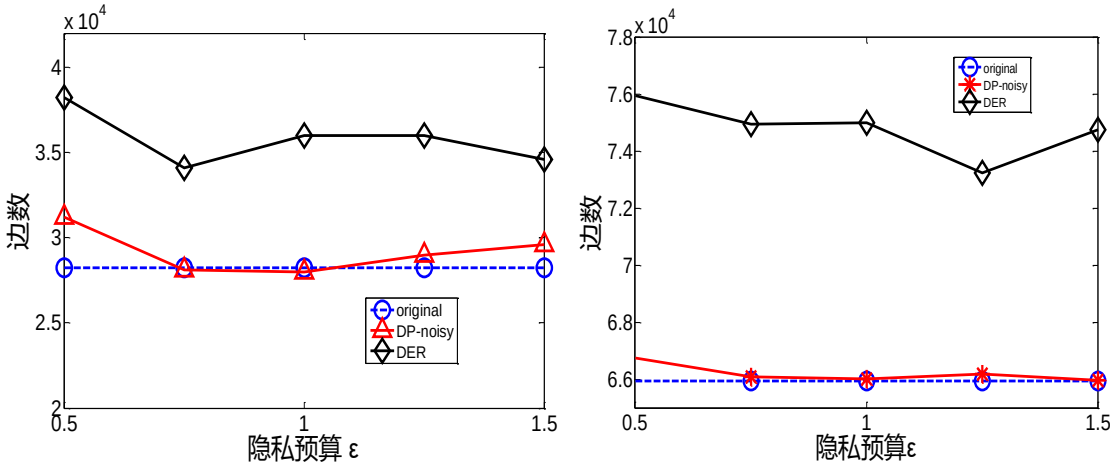
图 5.2　聚集系数对比

如图 5.2所示，当选择不同的隐私预算时，使用这两种方法的聚集系数都不会发生大的改变，其中使用 dp-noisy 方法的聚集系数大多数时候略高于原始数据，因为 dp-noisy 方法会在两个互相可到达但是不存在边的节点对之间添加边关系，同时选择度较低的节点为基准点插入虚假节点，会在一定程度上增加图的聚集程度，而 DER 方法则是通过聚类之后添加边噪声，这在一定程度上减少了对聚集系数的影响。

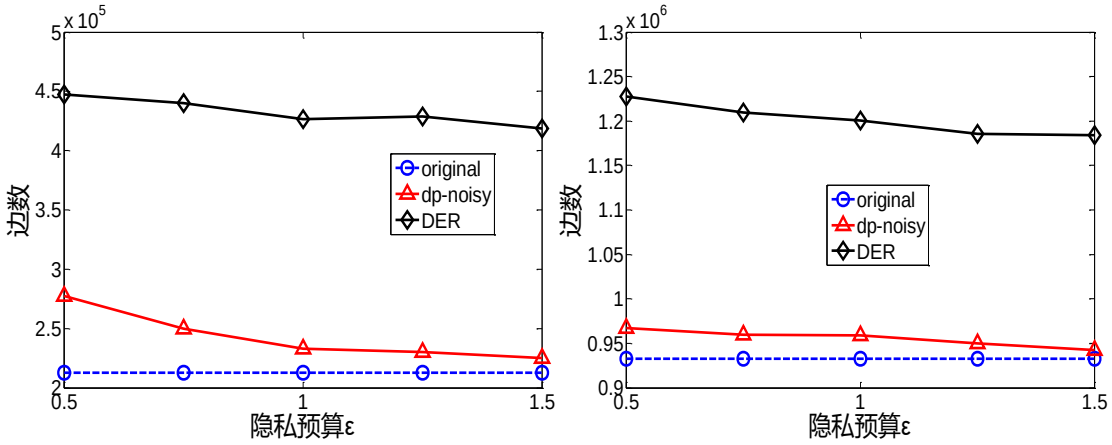
对比图 5.2的(a)和(b)还可以发现在更大规模的数据集上，两种方法对聚集系数的影响将会减小。因为对于稀疏矩阵而言，更大的数据集意味着噪音边有着更大的概率被添加到无效区域，从而对聚集系数的影响更小。

5.3.2 边数对比

图 5.3的(a),(b),(c)和(d)分别是数据集 1，2 和 3，4 在 dp-noisy 算法下和 DER 算法下边数的变化对比。



(a) data1 的边数　　(b) data2 的边数



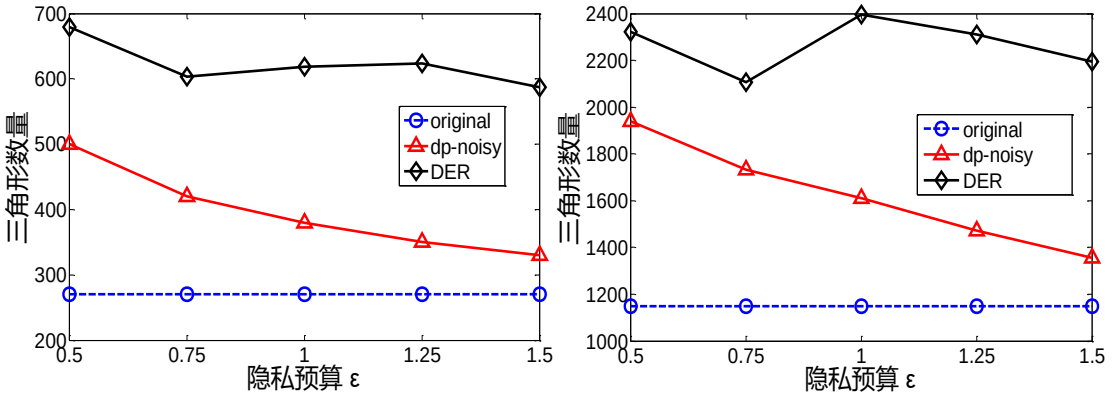
(c) data3 的边数　　(d) data4 的边数

图 5.3　边数对比

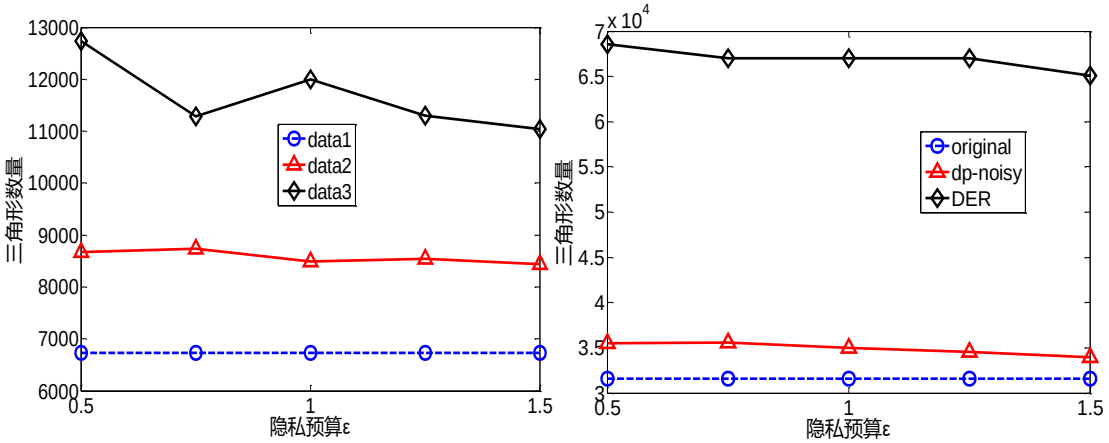
图 5.3是每个数据集上，通过两种算法扰动之后的边数量对比，可以看到扰动后的边数量往往会高于原始数据。对比图 5.3(a), (b), (c)和(d)可以发现，当数据集增大时 DER 算法的数据可用性越来越差，虽然 dp-noisy 方法也会增加边数量但影响远小于 DER 方法。这是因为 DER 通过添加大量的边噪音以达到保护效果，其中有相当一部分的冗余噪音边严重的破坏了数据的可用性。

5.3.3 三角形数对比

图 5.4的(a), (b), (c)和(d)分别是数据集 1，2 和 3，4 在 dp-noisy 算法下和 DER 算法下三角形数的变化对比。



(a) data1 的三角形数　　(b) data2 的三角形数



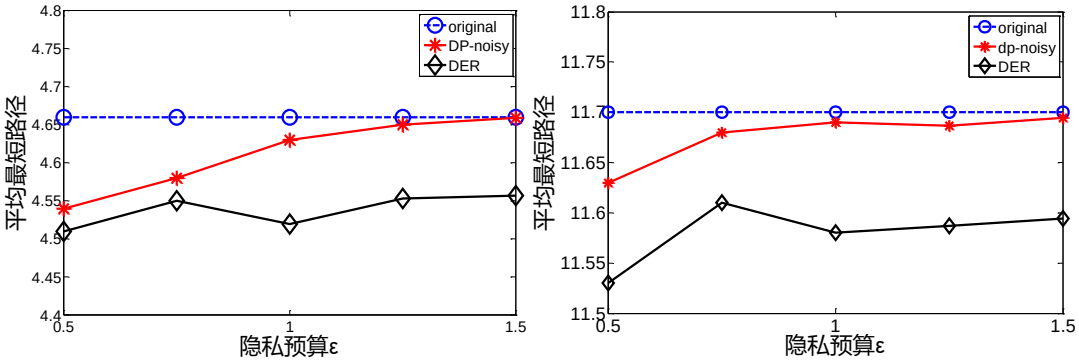
(c) data3 的三角形数　　(d) data4 的三角形数

图 5.4　三角形数对比

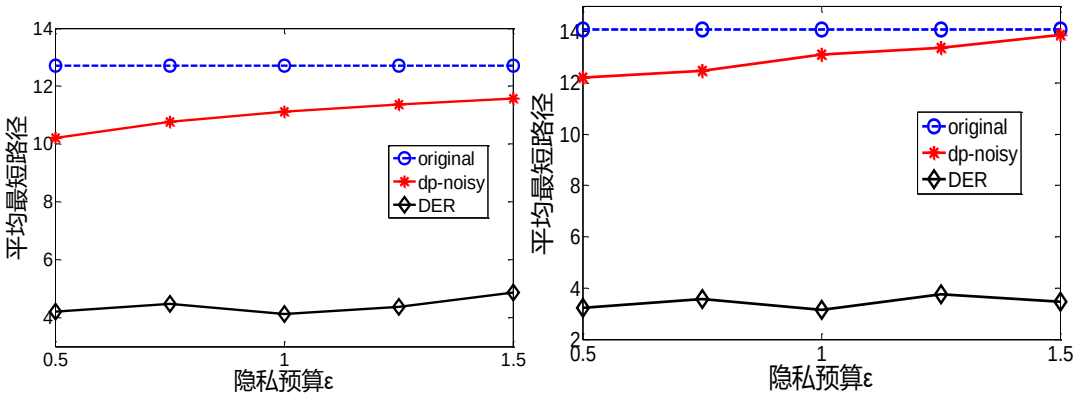
由于边数量的增加，三角形数量也在随之增加。如图 5.4所示，DER 方法产生的大量噪音边使得扰动后的三角形数量急剧增加，尤其是在大规模数据集 data3 和 data4 中，如图 5.4(c)和(d)所示 DER 方法扰动后的三角形数量超过原数据的两倍，这是因为 DER 方法需要进行聚类划分，在社区密集处添加扰动边对三角形数量的影响要大于非密集处。对比与 DER，本方法则可以在隐私预算取较大值时达到很好的数据可用性。

5.3.4 平均最短路径对比

图 5.5的(a), (b), (c)和(d)分别是数据集 1，2 和 3，4 在 dp-noisy 算法下和 DER 算法下平均最短路径的变化对比。



(a) data1 的平均最短路径　　(b) data2 的平均最短路径



(c) data3 的平均最短路径　　(d) data4 的平均最短路径

图 5.5　平均最短路径对比

如图 5.5所示，边数的增加除了会影响三角形数量之外，平均最短路径也会因为边的增加而减小。当隐私预算取值较大时，本算法扰动后的平均最短路径与原数据很接近，而 DER 方法则因为聚类之后引入大量的噪音边导致平均最短路径长度低于原数据。如图 5.5(c)和(d)所示，DER 方法对平均最短路径的影响和三角形数量一样，在大规模数据集上数据可用性也会变得更加糟糕。

综上所述，本方法与 DER 方法相比，在相同的隐私预算下有更好的数据可用性，尤其是在大规模的数据集上也可以实现较好的效果，并且解决了现有带权图扰动方法无法抵御结构攻击的问题。

5.4 扰动效果分析

针对边权值的分析，生成的扰动图权值及其属性要保持不变，需满足约束条件尽可能完整，约束不等式的数量在一定程度上决定了最终约束的完整性。下表记录了本实验在数据集 data1，data2，data3，data4 上每个阶段产生的约束不等式数。

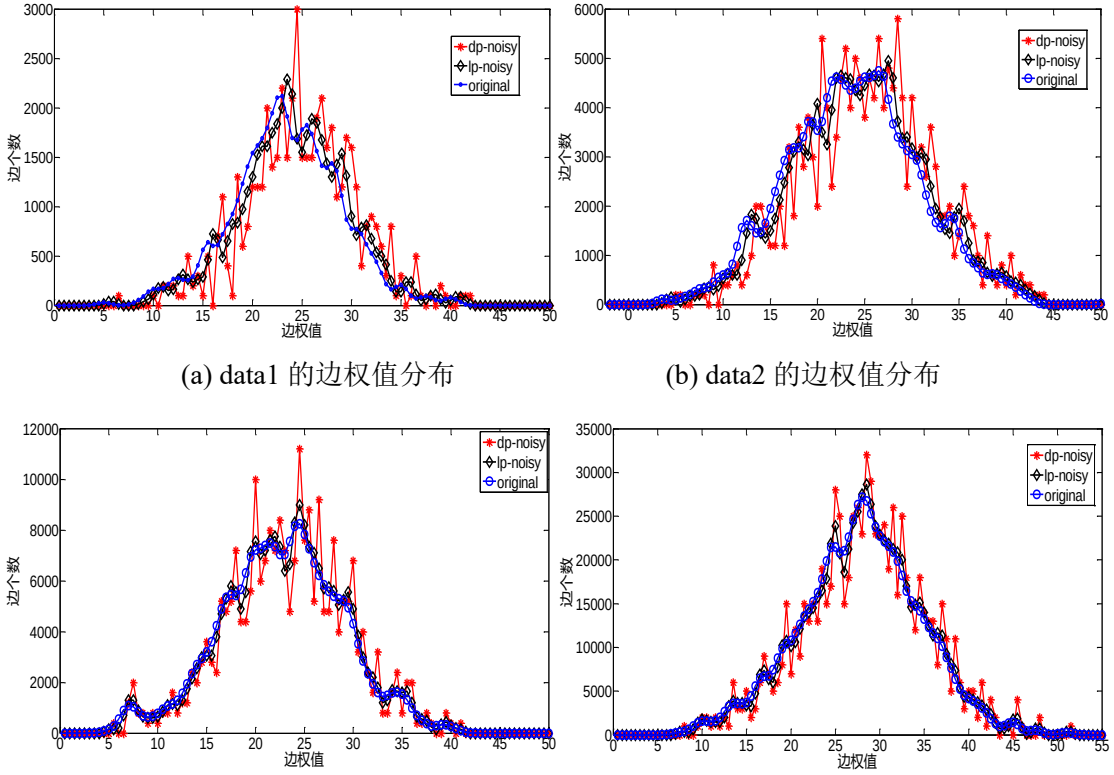
表 5.3　约束不等式数量表

数据集	约束不等式 1	约束不等式 2	约束不等式总和	边数
data1	28351	997	29348	28247
data2	94774	4961	99735	65940
data3	215608	9924	225532	212781
data4	935933	24874	960807	932974

由表 5.3 可知，本方案执行过程中集合 A 中的约束不等式数量大于图中的边数量，则可认为本方法得出的边权值属性与原图保持一致。

对于权值的隐私保护性分析，本文通过对比扰动前后边权值的分布进行。本实验选取了[2]中的权值扰动方法 lp-noisy 分别在四个数据集上进行比较，其中

dp-noisy 的隐私预算 ε 取值为 1，分配方式为 $\varepsilon_1:\varepsilon_2:\varepsilon_3=2:1:2$。图 5.6的(a),(b),(c)和(d)分别是数据集 1，2 和 3，4 在 dp-noisy 算法下和 DER 算法下边权值的分布对比。



(a) data1 的边权值分布　　(b) data2 的边权值分布

(c) data3 的边权值分布　　(d) data4 的边权值分布

图 5.6　分布对比

如图 5.6所示，本方法 dp-noisy 的扰动效果明显高于 lp-noisy 的扰动效果。数据规模较小时，如图 5.6(a)和(b)所示，lp-noisy 的扰动效果还比较明显，是因为 lp-noisy 方案通过线性规划方法求解单源最短路径上的边权值，并且只对不在路径上的权值加噪，因此对于较低的权值并没有很好的扰动。如图 5.6 (c)和(d)所示，当数据规模较大时用 lp-noisy 扰动与原数据差异很小，尤其在权值极大和极小处，lp-noisy 扰动后分布与原始数据几乎一致，出现这种情况是因为数据规模大时，边数的增加会引起约束集的增加，这使得线性规划求出的最优解就是原始数据。而本方法，除了对不在最短路径上的边权值扰动之外，还对线性规划的解添加差分隐私噪声，同时本方法在节点扰动过程中也对权值分布产生了较大改变，在图 5.6 (a)(b)(c)(d)中的可以发现，本方法总能产生非常明显的扰动。

第六章　总结与展望

6.1 总结

随着信息网络的不断发展，可以生成大量的网络数据并执行广泛的数据分析任务。随着超大规模社交网络的出现，需要在大量的社交关系中划分出不同敏感程度的边关系，并为这些边关系赋予不同的权值。相比于无权值的社交网络处理方法，能有效减少需要扰动的边数量。差分隐私是为表格数据开发的隐私标准，它提供了强大的隐私保护，而无需考虑攻击者拥有多少背景。然而，差分隐私最初是为表格数据设计的，并不适合社交网络的图表。

本次毕业设计中，设计并实现了基于差分隐私保护机制的隐私保护方法和相关的原型系统，该系统满足数据的输入、数据加密、效果监测及数据发布等需求。其中隐私保护方法在第三章进行了详细描述并且在第五章对算法性能进行了分析，可以看出本算法能够针对用户不同的需求，通过调整隐私预算以及需要保护的图属性选择合理的扰动方式，降低对数据可用性的影响。同时第四章介绍的隐私保护原型系统与隐私保护算法可以完美结合，用户可以通过输入文本格式的图三元组数据，本系统可以解析出原始图数据，再通过调用第三章的隐私保护方法对图进行加噪扰动，最后输出扰动后的图数据，并以文本格式返回给用户。

以上为本次毕业设计的创新特色，设计的原型系统具有优秀的交互性，同时算法也具有较好的数据可用性以及较好的隐私保护性能。因此，本次课题的研究对于解决数据拥有者在使用数据时产生的隐私问题有着较高的意义与价值。

6.2 展望

本次毕业设计研究中，虽然设计的原型系统以及隐私保护方法在隐私保护上取得了较好的效果，但是仍然存在许多不足之处可以进行改进，但由于毕业设计周期较短未加以实现。

大规模数据的优化处理，在大数据时代，各企业尤其是大型互联网公司，若要通过本系统进行隐私保护耗时将是一个严峻的问题，因此可以在数据输入时进行数据分割处理，将原始图拆分成不同小块再进行后续步骤。显然对于大规模数据而言，即当 N 足够大时，4 个 N^2 规模的图运行时间要远小于 $4N^2$ 规模的图。至于如何对图进行分割和分割后结果的整合需要进一步的研究。

另外，虽然当下隐私保护方法种类繁多，但是国内外的用户隐私泄露问题频发，这从一定程度上说明隐私保护方法并不能很好的解决现实中的问题。本次毕设研究中虽然提供了这一系列保护方法，隐私保护问题仍需要政府、企业以及用户本身等各方面的共同努力积极承担相应责任。

结束语

伴随着这篇论文进入尾声，我的大学生活也即将告一段落。回首过去的半年，从 2017 年 12 月 15 日至今，我一直沉浸在毕业设计的研究当中，从全面了解社交网络相关技术，掌握数据隐私保护的相关知识，到学习差分隐私和其它隐私保护方案的基本原理和工作机制，再到学习信息安全、数据库、密码学、数据结构和算法分析的基本技术和原理，包括练习使用 MATLAB 仿真平台以及 Java 的 web 开发，每一步都是对自己固有知识的一种新的突破。

在过去的这半年里，我收获的不仅仅是专业上的知识，更是一个大学生对待学术应该有的态度：如何在一筹莫展的时候制定学习计划，在面对困难的时候如何寻找解决办法，如何与指导老师沟通寻求帮助，如何将脑中的思路转化为文字。这些不仅对我接下来的研究生学习大有帮助，甚至对我将来的工作生活都大有裨益。

毕业设计的结束并不代表着研究的结束，这次的毕业设计已经为我打开了社交网络隐私保护的大门，我将在接下来的日子里继续对社交网络隐私保护技术继续展开研究。

致　谢

我要感谢我的毕业设计指导老师黄海平教授，从日常的学习、论文题目的确定等方方面面都给予我指导和督促，正是在他的关怀和帮助下，我的毕业设计才能如期完成。而他严谨细致的态度，春风化雨般的教诲也将一直影响着我，成为我的榜样。

同时，在毕业设计方案实现过程中，张东军学长也给予了我不小的帮助，在此对学长表达感谢。

最后，我还要向一直以来默默帮助我的父母表达感谢，正是他们无微不至的关怀与理解，给予了我一路向前坚持下去的动力。

参考文献

[1] 熊平, 朱天清, 王晓峰. 差分隐私保护及其应用[J]. 计算机学报, 2014, 37(1):101-122.

[2] Das S, Ömer Eğecioğlu, Abbadi A E. Anónimos: An LP-Based Approach for Anonymizing Weighted Social Network Graphs [J]. IEEE Transactions on Knowledge & Data Engineering, 2012, 24(4): 590-604.

[3] Zhou B, Pei J. Preserving Privacy in Social Networks Against Neighborhood Attacks[C]// IEEE, International Conference on Data Engineering. IEEE, Washington, DC, USA 2008: 506-515.

[4] Dwork C. Differential privacy[C]// Proceedings of the 33rd International Colloquium on Automata, Languages and Programming(ICALP'06). Venice, Italy, 2006: 1-12.

[5] Chen R, Fung B C M, Yu P S, et al. Correlated network data publication via differential privacy [J]. Vldb Journal-the International Journal on Very Large Data Bases, 2016, 23(4): 653-676.

[6] Zou L, Chen L, Zsu M T. k-automorphism: a general framework for privacy preserving network publication [J]. Proceedings of the Vldb Endowment, 2009, 2(1): 946-957.

[7] 张啸剑，孟小峰. 面向数据发布和分析的差分隐私保护[J]. 计算机学报，2014，37(4)：927-949

[8] 霍峥, 孟小峰. 一种满足差分隐私的轨迹数据发布方法[J]. 计算机学报, 2018, 41(02): 400-412

[9] Dijkstra E W. A note on two problems in connexion with graphs [J]. Numerische Mathematik, 1959, 1(1): 269-271.

[10] Sala A, Zhao X, Wilson C, et al. Sharing graphs using differentially private graph models[C]// ACM SIGCOMM Conference on Internet Measurement Conference. ACM, Berlin, Germany, 2011: 81-98.

[11] Liu L, Wang J, Liu J, et al. Privacy preserving in social networks against sensitive edge disclosure [R]. Technical Report CMIDAHiPSCCS006-08. Department of Computer Science, University of Kentucky, KY, 2008.

[12] Rathore S, Sharma P K, Loia V, et al. Social network security: issues, challenges, threats, and solutions[J]. Information Sciences, 2017, 421:43-69.

[13] Casas-Roma J, Herrera-Joancomartí J, Torra V. k -Degree anonymity and edge selection: improving data utility in large networks[J]. Knowledge & Information Systems, 2016, 50(2): 1-28.

[14] Xiao Q, Chen R, Tan K L. Differentially private network data release via structural inference[C]// Acm Sigkdd International Conference on Knowledge Discovery & Data Mining. ACM, New York, USA, 2014: 911-920.

[15] Wang Y, Wu X. Preserving Differential Privacy in Degree-Correlation based Graph Generation.[J]. Transactions on Data Privacy, 2013, 6(2): 127-145.

[16] 王俊丽, 管敏, 魏绍臣. 面向社交网络分析的差分隐私保护研究综述[J]. 高技术通讯, 2015, 25(3):239-248.

[17] 吴英杰. 隐私保护数据发布: 模型与算法[M]. 北京: 清华大学出版社, 2015.

[18] Gao T, Li F, Chen Y, et al. Preserving Local Differential Privacy in Online Social Networks [J]. 2017, pp. 393-405.

[19] Zheleva E, Terzi E, Getoor L. Privacy in Social Networks[C]// USA: Morgan & Claypool Publisher, 2012:85.

[20] Hay M, Li C, Miklau G, et al. Accurate Estimation of the Degree Distribution of Private

Networks[C]// Ninth IEEE International Conference on Data Mining. IEEE Computer Society, 2009:169-178.

[21] Mir D J, Wright R N. A Differentially Private Graph Estimator[C]// IEEE International Conference on Data Mining Workshops. IEEE, 2009:122-129.

[22] Leskovec J, Faloutsos C. Scalable modeling of real graphs using Kronecker multiplication[C]// Machine Learning, Proceedings of the Twenty-Fourth International Conference. DBLP, 2007:497-504.

[23] Dwork C. Differential Privacy: A Survey of Results[M]. Theory and Applications of Models of Computation. Springer Berlin Heidelberg, 2008:1-19.

[24] Dwork C, Mcsherry F, Nissim K. Calibrating Noise to Sensitivity in Private Data Analysis[M]// Theory of Cryptography. Springer Berlin Heidelberg, 2006:265-284.

[25] Mcsherry F, Talwar K. Mechanism Design via Differential Privacy[J]. 2007:94-103.

[26] Rathore S, Sharma P K, Loia V, et al. Social Network Security: Issues, Challenges, Threats, and Solutions [J]. Information Sciences, 2017, 421:43-69.

[27] Li X, Yang J, Sun Z, et al. Differential Privacy for Edge Weights in Social Networks [J]. Security & Communication Networks, 2017, 2017(4):1-10.

[28] Gao T, Li F, Chen Y, et al. Preserving Local Differential Privacy in Online Social Networks [J]. 2017, pp. 393-405.

[29] R Chen, B C Fung, P S Yu, B C Desai. Correlated network data publication via differential privacy [J]. Vldb Journal -the International Journal on Very Large Data Bases, 2014, 23(4): 653-676.