有指导的图抽样技术

Yousuf, Muhammad Irfan and Kim, Suhyun

摘要

随着数据采集技术的蓬勃发展以及图数据的广泛应用, 对大图进行运算分析是一项重要工作。但是, 由于图结构的复杂性以及现在极大的图规模, 对全图进行精确计算的效率底下且成本极高。一个可以解决该问题的简单想法便是近似计算技术, 即选择图的部分代替全图进行计算。那选择图的哪一部分代替全图可以使得分析的准确性更高呢? 专门研究该问题的图抽样技术顺应而生。传统的图抽样技术现今已有非常多的改良版本, 但是有一个通病, 就是要想获得良好的抽样效果就需要 10% 以上的抽样比例, 低于 10% 抽样效果会迅速降低。本工作所复现的论文提出了一个这样的想法若在抽样之前已知图的部分信息, 可以利用已知信息指导抽样过程, 该论文提出了一个用平均聚类系数与平均度图属性指导的图抽样算法 Guide Sampling, 并且实验证明了该算法可以将抽样比例降至 1% 以下。

该算法没有公开代码, 因此, 本工作对 Guide Sampling 算法利用 Python 复现, 而且由于 R 有强大的图分析 library, 因此也利用 R 进行了复现以方便他人使用 R 进行抽样且分析。本工作把 Guide Sampling 与传统的抽样算法在 1% 进行效果比较, 以验证论文所述的效果, 而且同时探讨了若指导抽样的图属性值有误对抽样效果的影响。从结果可以发现,Guide Sampling 在一般大图上抽样效果与传统算法差不太多, 但是这样利用先验知识指导图抽样的想法值得学习与发展。

关键词: 图抽样; 图属性估计; 先验知识

1 引言

随着物联网的迅猛发展, 越发精良和普及的采集终端无处不在地为我们把人类无时无刻创造海量的数据采集收集起来, 大数据作为第三次信息化浪潮的标志之一, 从被提出至今一直受到各个领域的重视[1]。图数据除了展示个体本身 (节点), 还能表示出个体之间的联系 (边), 能更好的反映出现实世界特征, 因此, 在现今各个领域十分常用, 超大图数据分析成为了现今大数据分析的重要部分。面对庞大而复杂的真是世界图数据, 精确计算的成本高效率低下的问题难以被忽视, 大量的内存消耗和不足的计算能力使得在大图分析中频繁使用精确计算是不切实际的。

一个大数据分析中解决计算技术不足的思路就是使用近似计算技术, 对于图数据, 即为了理解超大真实世界图的结构和性质, 可以从一个大图中提取大图中的部分节点与边构成的样本子图代替大图进行计算, 想要获得尽量准确的分析结果, 子图当然需要尽可能的保留大图拓扑属性和特征。一个简单的共识, 在抽样技术中, 抽样比例越高自然抽样的效果会更好, 现有的经典图抽样算法, 特别是基于遍历的抽样算法, 抽样比例在 10% 以上才能获得相对良好的效果, 低于该比例抽样效果显著下降, 尽管已经降低了 90% 的图规模, 但是在超大图当中, 譬如全图有 10^{10} 个节点时, 意味着传统抽样获得的样本子图大小为 10^9 个节点, 样本子图的规模使得抽样失去意义, 这听起来很夸张, 但是 Facebook 今年 10 月月活跃用户突破 20 亿。这使得对此类图结构的分析变得耗费巨, 而论文 Guided sampling for large graphs[2]的目标是在保持图的结构和关键属性的同时, 产生样本量低至 0.1% 的良好样本。实际上, 在许

多情况下进行图分析之前, 分析者是可以知道一些图相对准确的信息, 将其称为" 先验知识", 如何有效利用这些知识指导图抽样过程是该论文的目的。本工作对上述论文所提出的抽样算法进行复现以及分析。

2 相关工作

2.1 图数据

Definition 1. 图 ***(Graph):***

存在一个二元组 $G=(V,E)$, 其中 V 是 G 点集,$V=\{v_i|i=1,2,....N\},|V|=N;V\times V\rightarrow E,E$ 是 G 的边集,$|E|=M,G$ 被称为图。

若 $E=\{(v_i,v_j)|v_i,v_j\in V\},(v_i,v_j)$ 是无向数对,E 是无向图 G 边集; 若 $E=\{<v_i,v_j>|v_i,v_j\in V\},<v_i,v_j>$ 是有向数对,E 是有向图 G 边集。常用的表示图结构的方式有邻接矩阵及边对, 如图 *1*所示。本工作用于实验输入的图数据是边对形式。

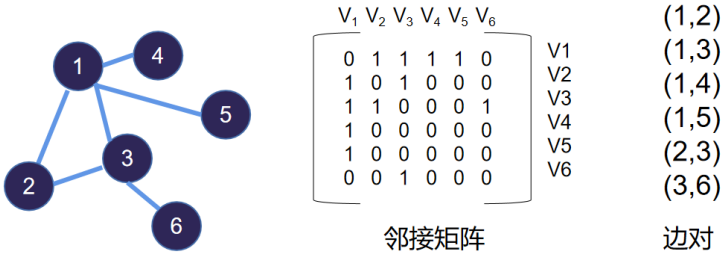

图 1: 图数据

大型图数据类型现在普遍分成两种, 第一种是许多小图的大集合, 称为事务图, 这种图通常用于生物或者化学研究领域, 例如[3]中提到的在 DBLP 网络中, 每个用户都与自己发表的论文集合构成小图结构; 还有生物学中, 每个小图表示一种特定的氨基酸结构, 许多氨基酸结构共同构成蛋白质结构; 另一种是包含数十亿个顶点和边的单一高连通大型图, 这类图通常用于为社交网络、业务分析等方面, 本研究主要研究的是就是所述的后一种图结构, 即高连通度大图数据。

2.2 图抽样技术及图属性

图抽样实际上是一种选择边集点集对应子集构成一个新图的一种图变换[4], 一般图抽样适用于两种情况: 在全图未知的情况下, 通过研究子图对全图进行探索; 在全图已知的情况下, 为了获得一个与原图 "相似" 但更小规模的子图进行图计算, 从而从子图中可以得到原图的某些信息, 例如异常点、属性值。

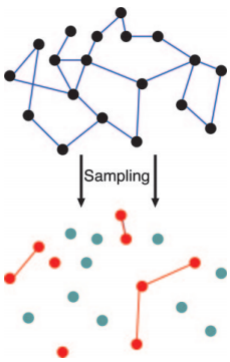

图 2: 图抽样

通过对现有的一些图抽样算法的研究进行分析[5-9], 一般图抽样目的是对原图进行属性值的估计, 部分图抽样研究专注于属性保存, 即通过抽样使得属性可以更大程度上在子图中保留。更多地研究致力于属性估计, 也就是说, 抽样算法不一定保留某些属性, 只要可以根据抽样估计出较准确的属性值, 就认为抽样算法是有用的。因此, 本工作的重点也是在对大图进行抽样过程中, 尽可能的保留原图信息, 使得样本子图在近似计算中准确性更高。

现今, 图数据抽样算法有许多种, 核心的三种是: 随机点抽样、随机边抽样、遍历抽样[10], 基于这三种算法有许多变形, 例如, 基于随机节点抽样的随机点伴随邻居抽样 (Vertex Sampling with Neighbourhood), 基于随机边抽样的带收缩的边抽样算法 (Edge Sampling with Contraction), 基于遍历抽样的滚雪球抽样算法 (Snow-Ball Sampling)。这些算法之间没有绝对的优劣之分, 在面对不同问题的时候有不同的表现效果。

⋆ 随机点抽样 (Vertex Sampling): 对于大图 $G = (V, E)$ 任意选取点集 $V_s \subseteq V$, 将 V_s 之间的所有边取出 $E_s = \{(v_i, v_j) \in E | v_i, v_j \in V_s\}, E_s \subset E$, 构成子图 $G_s = (V_s, E_s)$。

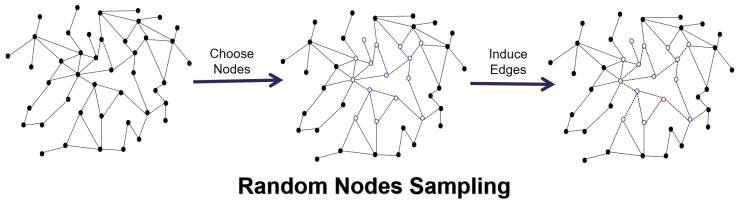

图 3: 随机点抽样

⋆ 随机边抽样 (Edge Sampling):

对于大图 $G = (V, E)$ 任意选取边集 $E_s \subseteq E$, 将 E_s 的所有关联节点取出 $V_s = \{v_i \in V | (v_i, v_j) \in E_s\}, V_s \subset V$, 构成子图 $G_s = (V_s, E_s)$。

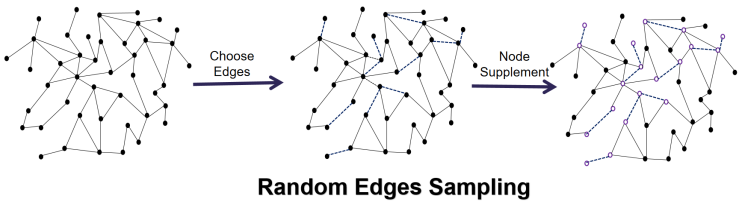

图 4: 随机边抽样

⋆ 遍历抽样 (Travel-based Sampling):

随机从图中抽取一个节点 $v_i \in V$, 以某种选定的遍历方法对图进行遍历, 把所有遍历过的边与节点构成子图 G_s。从 TBS 中抽取的样本有相关性, 但有时可以利用这一点来构建某些属性的更有效的估计。

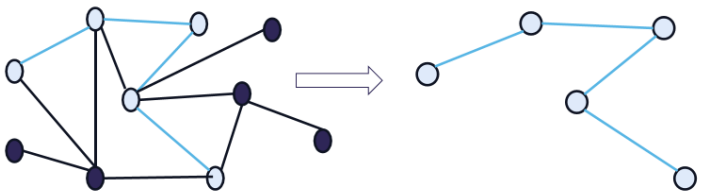

图 5: 随机点抽样

图抽样算法的评价是通过衡量子图与原图的“相似性”来进行, 而由于图结构的复杂性, 子图与原图的“相似”需要基于不同的图抽样目标来衡量, 例如, 若为了得到图中某些异常节点, 衡量标准可以

为子图涵盖了原图异常点的百分值, 若为了得到原图某属性, 衡量标准可以为子图估计的属性值与原图真实属性值的偏差。本研究的图抽样目的是尽可能保留原图属性, 因此在介绍图抽样算法的同时需要对图属性有所了解。图的属性可分为两类[10]: 分布型属性, 例如, 图的度分布, 联合度分布; 另一种是总结型属性, 例如, 图的密度, 平均度; 因此本论文为不同的图属性设计不同的“相似性”标准:

⋆ 单值型属性:

使用均方根误差 (Root Mean Square Error) 来衡量某单值型属性实际值与估计值的差距:

$$RMSE = \sqrt{\frac{1}{N}\sum_{1}^{n}(\theta_s - \theta_A)^2},$$

其中,θ_s 和 θ_A 分别为采样值和原始值,n 是估计次数。

⋆ 分布型属性:

使用 JS 散度[11](Jensen–Shannon Distance) 衡量属性在抽样子图上经验分布与原图经验分布的比较:

$$D_{JD} = \frac{1}{2}D_{KL}(P||M) + \frac{1}{2}D_{KL}(Q||M),$$

其中,P、Q 是两经验分布,$M = \frac{1}{2}(P+Q)$,D_{KL} 表示 KL 散度。

3 本文方法

由于在经典图抽样算法中, 当抽样比例 rate 较低时,G_s 太小, 无法捕获 G 的结构, 这是因为抽样是在没有任何基础原始图的先验知识的情况下盲目执行的, 本论文认为, 如果向抽样技术提供一些信息, 并将抽样引向已知目标, 可以生成 rate 较低的良好样本。

3.1 准备工作

有这样一个事实, 一般情况下在无需消耗大量资源的情况下, 全图度 (average degree) 和聚类系数 (average clustering coefficient) 的平均值可以准确有效地估计, 因此, 本论文希望可以使用这两个属性值来指导抽样过程, 以便提取微小样本的同时保留图的属性和结构特征。

本工作采取基于遍历的抽样方法, 该类抽样方法较随机节点和随机边抽样更适合实际应用, 因为它仅利用节点的局部信息, 而不是图的全局信息, 而且由于社交网络 (online social network) 的 API 限制, 一般不能获得全图信息, 因此一般社交网络的抽样只能采取遍历抽样, 这使得本工作的方法成为爬行社交网络的实用选择。

已知 MH 随机游走抽样算法 (Metropolis-Hastings Random Walk) 在度的平均值估计有效[12-13], 随机游走在聚类系数的平均值[14]估计有效, 因此论文先用 MH 随机游走与随机游走进行抽样估计这两个属性值, 如果是对应社交网络的服务商, 可以跳过这一步, 利用已知的用户信息直接输入这两个属性值。

Definition 2. 度 ***(Degree):***

对于无向图 $G = (V, E)$,$\forall v_i \in V$,v_i 的度定义为与该点邻接的边数目,

$$Deg(v_i) = |\{(u, v)|u = v_i, v \in V, (u, v) \in E\}|,$$

由图 $G = (V, E)$ 所有点度值总体形成的分布称为图 G 的度分布 *(Degree Distribution)*。

平均度 ***(Average Degree):***

$$AD = \frac{\sum_{v_i \in V} Deg(v_i)}{|V|} = E[deg].$$

Definition 3. 聚类系数分布 ***(Clustering Coefficient distribution):***

局部聚类系数用来描述一个顶点的邻居之间结集成团的程度的系数, 对于任一结点 $v_i \in V$ 若 $degree(v_i) \geqslant 2$,$N(v_i)$ 表示 v_i 的邻接点的集合,$K = \{(u,v)|u,v \in N(v_i)\}$。

点 v_i 局部聚类系数 $C(v_i)$ 定义为 $N(v_i)$ 之间存在的边数目与 $N(v_i)$ 之间最多的边数目比值[15]:

$$CC(vi) = \frac{|K|}{\binom{|N(Vi)|}{2}},$$

图 G 中所有度大于 2 节点的局部聚类系数形成的分布为聚类系数分布。

平均聚类系数 ***(Average Clustering Coefficient[16]):***

$$ACC = \frac{\sum_{v \in V} C(v)}{|V|}.$$

3.2 Guided Sampling

3.2.1 方法概述

获得平均度与平均聚类系数两个属性后开始抽样工作阶段一利用修正的 Depth First Sampling 抽样获得一个节点数目符合要求, 但是边过抽样的抽样子图 $G_s = (V_s, E_s)$; 阶段二对阶段一获得的 G_s, 在已获得的平均度与平均聚类系数的指导下, 对过抽样的 E_s 进行删减, 为 E_s 计算权值, 按照指导选择对应边进行删除。

本工作利用属性平均度与平均聚类系数指导抽样, 获得一个大幅度降低规模的子图, 同时保留其关键属性度分布 (degree distribution)、聚类系数分布 (clustering coefficient distribution)、路径长度 (path length)、直径 (diameter), 由于平均度与平均聚类系数容易获得, 且最终抽样子图规模较传统方法大幅度下降, 从而使得大图分析过程消耗资源大量减少。

3.2.2 边权重定义

节点的局部聚类系数衡量其邻域的连通性, 节点的相邻节点之间的现有边与相邻节点之间所有可能边的数量之比, 假设节点 v, 它的度为 $deg(v)$, 若 v 的邻接点之间移除了一条边, 那么它的聚类系数将降低 $\frac{2}{deg(v) \times deg(v)}$, 也使得全图平均聚类系数同等程度降低, 论文基于这一想法为边定义权值。

对于边 (u,w), 移除该边使得点 v 的聚类系数降低 $\frac{2}{deg(v) \times deg(v)}$, 因此点 v 赋予边 (u,w) 的权值为

$$W^v_{(u,w)} \frac{2}{deg(v) \times deg(v)},$$

按照这样的思路, 所有与 (u,w) 可以形成三角形的节点都能类地给 (u,w) 赋予权值, 将这些点的所有权值求和就能得到该边的权重

$$W_{(u,w)} = \sum_{v_i \in V_{(u,w)}} \frac{2}{deg(v_i) \times deg(v_)},$$

其中,$V_{(u,w)}$ 是能与边 (u,w) 形成闭合三角形的点集。

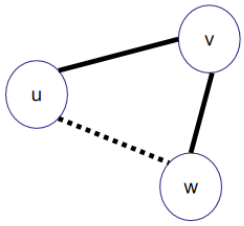

图 6: 权值定义

直观地可以认为, 具有高权重的边对图的平均聚类系数具有高影响, 因为, 高权重的边要么是能与很多节点构成三角形, 要么是可构成三角形的节点度值低, 低度节点对边缘的权重贡献更大, 并且当移除高权重边缘时, 低度节点的局部聚类系数显著降低, 从而导致图的平均聚类系数急剧下降, 但是权重与全图平均聚类系数之间没有必然数值联系, 仅仅是一个定性分析的结果。

3.3 复现细节

复现时候需要关注的细节

(1) 权值大于 0 的点才可以删除, 否则会带来孤立节点, 同样的, 只有当两个节点度数都大于 1 时, 才删除边, 这将确保没有孤立的节点。

(2) 实际上阶段一的 G_s 由于含有超量边数目, 因此 Avg-cc 必定高于原图, 因此在删边时候可以选择三种方案

- ⋆ 删除少量高权边;
- ⋆ 删除大量低权边;
- ⋆ 删除低权与高权边数目均衡, 本复现工作采用此种方法。

(3) 可以在删边前知道需要删除数目: 删边前的边为 $|E_s|$ 条, 平均度为 $d_{init} = \frac{|E_s| \times 2}{|V_s|}$, 原图的度为 V_org, 那为了使得抽样子图的度与原图一致, 要求子图的边数应该是 $\frac{d_{org} \times |V_s|}{2}$, 因此, 在点数不变情况下, 需要删除的边数目为

$$e_{extra} = |E_s| - \frac{d_{org} \times |V_s|}{2}.$$

(4) 选择 ModDFS 而非 DFS 的原因: 可以获得更多边以便进一步抽样; 选择 DFS 而非 BFS(Breadth First Search Sampling) 的原因: 可以获得直径更大的样本子图与原图更好匹配。

在 (2) 里面选择了可以删除低权边和高权边的方案, 为了决定每一轮迭代是移除低权重边缘还是高权重边缘, 该论文提出一个简单的方法假设删边前图的原聚类系数 c_{init}, 原图的聚类系数 c_{org} 是工作希望删边后聚类系数的目标值, 需要删除的边数目是 e_{extra}, 如果每条被删边对平均聚类系数影响是一样的话, 那么图的聚类系数下降趋势就是一条直线, 这条直线有斜率 $scope$ 刚好就是每删除一条边对平均聚类系数的影响, 通过这个 $scope$ 就可以计算出每条边被删后的期望下降到的值 c_{exp}, 如果删边后真实的平均聚类系数值大于期望值, 那么就要往高权值去删以希望真实的平均聚类系数值尽量等于期望值, 反之亦然。利用这个简单的想法, 当需要聚类系数急剧下降时, 移除了高权重边缘, 而如果需要小调整以遵循假设的线, 则移除了低权重边缘, 并在移除所有额外边缘后最终达到目标值。

3.4 与已有开源代码对比

本工作在实现图的深度遍历的时候参考了 python 中 littleballoffur 库的源代码, 本工作借鉴该库, 同样利用栈结构实现图的修正深度遍历抽样 (ModDFS), 其他均直接由论文给出的方法自己实现。本工作中部分图基本操作, 例如, 计算图节点数、计算图边数等操作, 直接调用 python 中 nxnetwork 库的相关函数; 本工作部分数学运算直接调用 numpy 库中函数, 部分随机算法调用 random 库函数, 部分队列或者栈操作调用 queue 函数。

R 语言具有多样的图分析工具, 因此本工作同样使用 R 语言实现了 Guide sampling, 用的是 igraph 包内含的图结构进行实验, 也使用 igraph 包的函数进行原图与抽样图的属性计算及比较。

Guided-sampling 没有开源代码, 在过程中需要实现内容包括

(1) 基于深度搜索的修正图深度遍历抽样算法 (ModDFS), 如算法 1所示;

(1) 基于边诱导算法, 如算法 2所示, 在实际 Guided Sampling 实现中, 将 ModDFS 与 inducetion 合并为了 ModDFS_indece 函数;

(2) 基于先验知识指导的图抽样算法 (Guided Sampling), 如算法 3所示。

Procedure 1 Modified Depth First Sampling

Input: Graph $G = (E, V)$, sampling ratio ϕ;
Output: sampled graph vertex set V_s;
Initialization:
Stack$S = \emptyset$,
$V_s = \emptyset$,
需要抽样数目 $num = |V| \times \phi$,
全图随机抽取一个起始点 v_start,
$S \leftarrow put.stack(v_start)$;

Sampling:
while $|V_s| < num$ **do**
  $source = get.stack(v_start)$ **if** $source \notin V_s$ **then**
    获得邻接点 $neighbors = get.neighbor(source)$,
    把邻接点按度的大小排序低度在前, 高度在后 $neighbor_{rank} = rank(neighbors.degree)$,
    按度从低到高顺序把邻接点入栈 $S \leftarrow put.stack(neighbor_{rank})$,
    把 source 加到 V_s 中 $V_s \leftarrow add(source)$,

return *sampled vertex set* V_s

Procedure 2 Inducetion

Input: Graph $G = (E, V)$, Sampled vertex set V_s;
Output: sampled edge set E_s;
Initialization: $E_s = \emptyset$;
for $e = (u, v) \in E$ **do**
  if $u \in V_s$ ***and*** $u \in V_s$ **then**
    $E_s \leftarrow add(e)$

return *sampled edge set* E_s

Procedure 3 Guided sampling

Input: Graph $G=(E,V)$,degree d_{org},clustering coefficient c_{org}, sampling ratio ϕ;
Output: sampled graph $G_s=(E_s,V_s)$;
Initialization:
$E_s=\emptyset, V_s=\emptyset$;

Over-sampling:
抽取子图点集 V_s ←Alg1 ModDFS(G,ϕ),
诱导获得过抽样子图边集 E_s ←Alg2 Inducetion(V_s,G),
构成过抽样子图 $G_s=(V_s,E_s)$;

Prepare for removing edges:
为 E_s 计算权重 $Weight(E_s)$,
为 E_s 按照权重 $Weight$ 降序排序,
计算需要删除的边数 $e_{extra}=|E_s|-\dfrac{d_{org}\times|V_s|}{2}$,
为 G_s 计算平均聚类系数 $c_init \leftarrow CaculateACC(G_s)$;

Removing extra edge:
记录删的边数目 $e_{del}=0$,
记录删的边比例 $e_{ratio}=1$,
记录现在的聚类系数 $c_{cur}=c_{init}$,
记录现在聚类系数与目标聚类系数的差距 $c_{ratio}=\dfrac{c_{org}}{c_{cur}}$,
计算假想直线斜率 $slope=\dfrac{\frac{c_{cur}}{c_{org}}-1}{e_{extra}}$,
计算此时平均聚类系数期望值 $c_{exp}=c_{init}-(slope\times e_{del}\times e_{ratio})$,
while $e_{del}<e_{extra}$ **do**
  $mid=\dfrac{|E_s|}{2}$,
  if $c_{cur}>c_{exp}$ ***and*** $c_{cur}>c_{org}$ **then**
    $index=mid\times c_{ratio}\times e_{ratio}$,
  else
    $index=mid+mid\times e_{ratio}$,
  按照索引删边 $Deleteedge(E_s,index)$,
  $e_{del}=e_{del}+1$,
  $c_{cur}\leftarrow CaculateACC(G_s)$,
  $c_{ratio}=\dfrac{c_{org}}{c_{cur}}$,
  $e_{ratio}=\dfrac{e_{extra}-e_{del}}{e_{extra}}$,
  $c_{exp}=c_{init}-(slope\times e_{del}\times c_{org})$,
return *sampled graph* $G_s=(E_s,V_s)$

4 实验结果分析

4.1 实验环境、界面分析与使用说明

本工作使用的电脑处理器是 $Intel(R)Core(TM)i7-10700CPU@2.90GHz2.90GHz$,Windows64 位操作系统, 平台是 python3.11, 编译器是 pycharm2022.2, 以及 R4.2.2, 编译器是 R Stuidio。

4.2 实验验证

本工作用 python 实现不同抽样抽样算法, 在 R 上实现抽样结果的比较。Guided Sampling 原论文选取的衡量属性集是：度分布、聚类系数分布、路径长度分布、图直径、同构性、平均路径长度这 6 个属性，同构性、平均路径长度是可以通过度分布与路径长度分布计算获得，本工作舍去。度分布、聚类系数分布、路径长度、直径上在抽样比例 1% 以下有较好的保留效果, 因此, 本工作使用 1% 的抽样比例对论文实验进行验证。分布属性度分布、聚类系数分布、路径长度分布, 使用 JS 散度。单值属性直径使用 RMSE 作为比较标准。度分布、聚类系数已给出定义, 其他属性定义如下所示。

Definition 4. 路径长度分布 ***(Path length distribution):*** 图中任意连通两点之间的最短路径长度形成的分布。

Definition 5. 直径 ***(Diameter):*** 图中连通两点最短路径的最长路径长度。

实验选择的图来自 KONECT[17], 如表 1所示。由于硬件的限制, 本工作选取的实验图的规模较原论文要小一个量级。

名字	简介	类型	节点 (个)	边 (条)
Random Graph	由随机生成 2000 × 2000 的对称 0-1 矩阵表示	随机生成图	2000	998719
MovieLens 100k	用户对电影评价关系图	社交网络图	2625	100000
Route views	相互连接的互联网自治系统	计算机关系网	6474	13895
Pretty Good Privacy	Pretty Good Privacy 算法的用户交互网络	用户网络关系网	10680	24316
Astrophysics	来自 arXiv 的“天体物理学”部分作者的的合著关系	合著关系网	16046	121251

表 1: 仿真模拟图的介绍

对 5 个图采用 Guided Sampling, 且输入准确的平均度与平均聚类系数进行指导, 得到的子图与原图在 3 个分布属性的对比如下图 7、8、9所示。红色是原图的分布, 蓝色是 Guided Sampling 获得的抽样子图的属性分布。

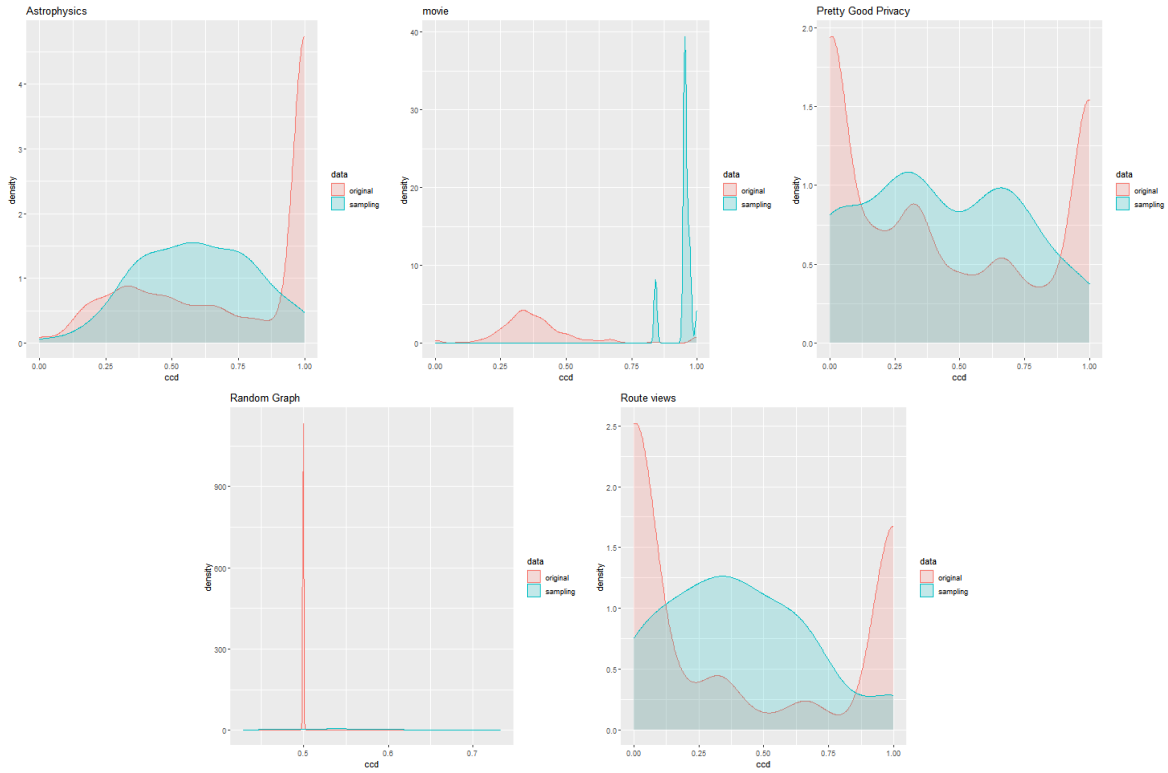

图 7: 聚类系数分布

从图 7对于两图聚类系数分布的比较可以发现, 抽样子图能部分获取原图的聚类系数属性信息, 但是原图的分布具有明显突变的地方没能获取, 比如 Astrophysics 原图聚类系数分布的尾部密度突然变高,Pretty Good Privacy 与 Route views 原图聚类系数分布的首尾突高, 抽样子图均没有捕获。而 Random Graph 是一个相对平均的分布, 但是抽样子图的分布却有突变。

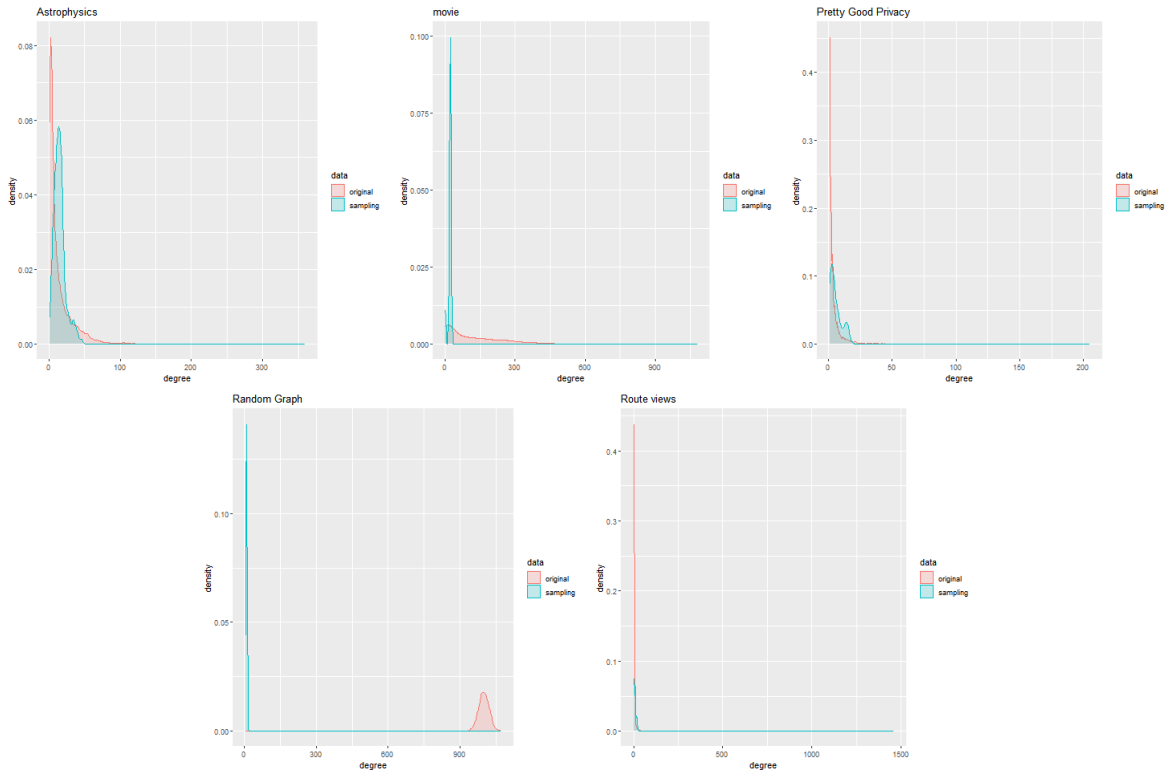

图 8: 度分布

从图 8对于两图度分布的比较可以发现, 抽样子图能部分获取原图的度属性信息, 而且效果好像比聚类系数分布要更好, 而且原图的度分布重尾的特征能被捕获。

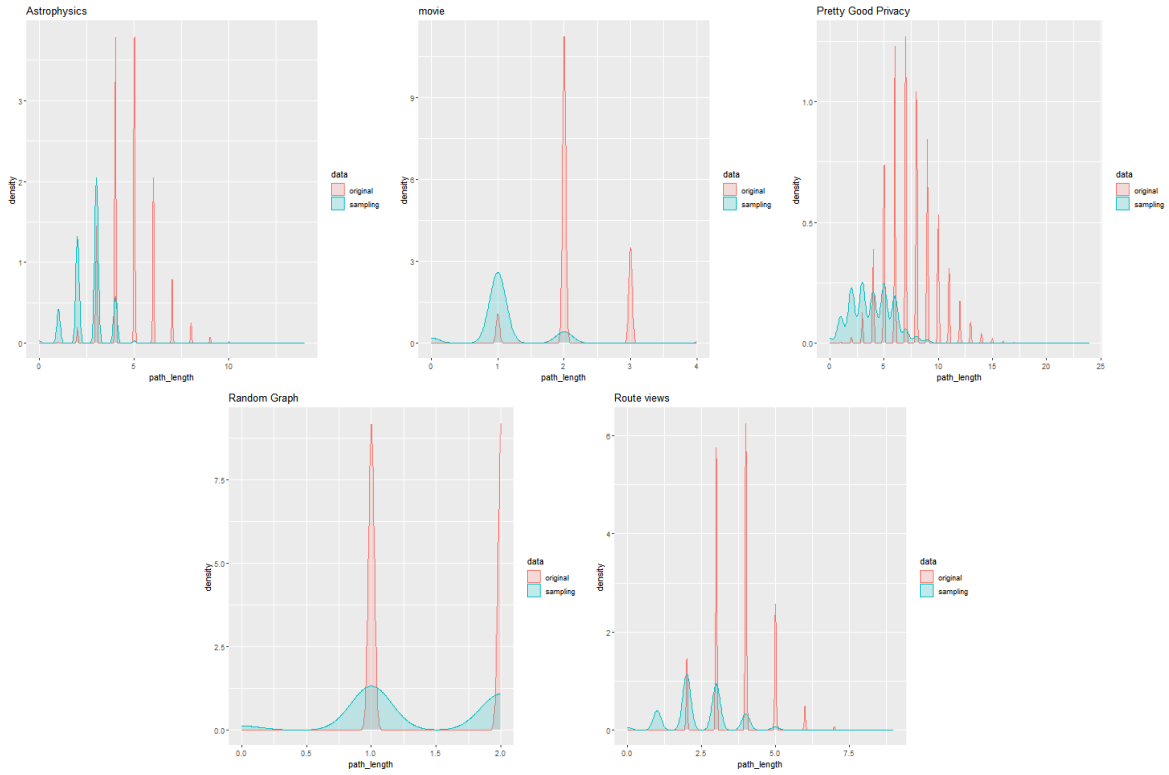

图 9: 路径分布

路径长度分布是离散分布, 从图 9对于两图路径长度分布的比较可以发现, 抽样子图能部分获取原图的路径长度属性信息, 但是从图看抽样效果一般, 抽样子图比原图的的分布要更集中, 一些长路径的节点对可能在抽样中被丢失。

4.2.1 与传统算法的效果比较

因为本论文的抽样是基于遍历的图抽样, 基于随机节点与随机边的抽样算法缺少意义, 因此, 本工作用于与 Guided Sampling 做比较的抽样算法是传统的三个遍历抽样算法 Random Walk sampling(RW),Forest Fire sampling(FF),Snowball sampling(SB), 这三个经典算法的实现使用了 Python 图抽样算法库 Graph_Sampling

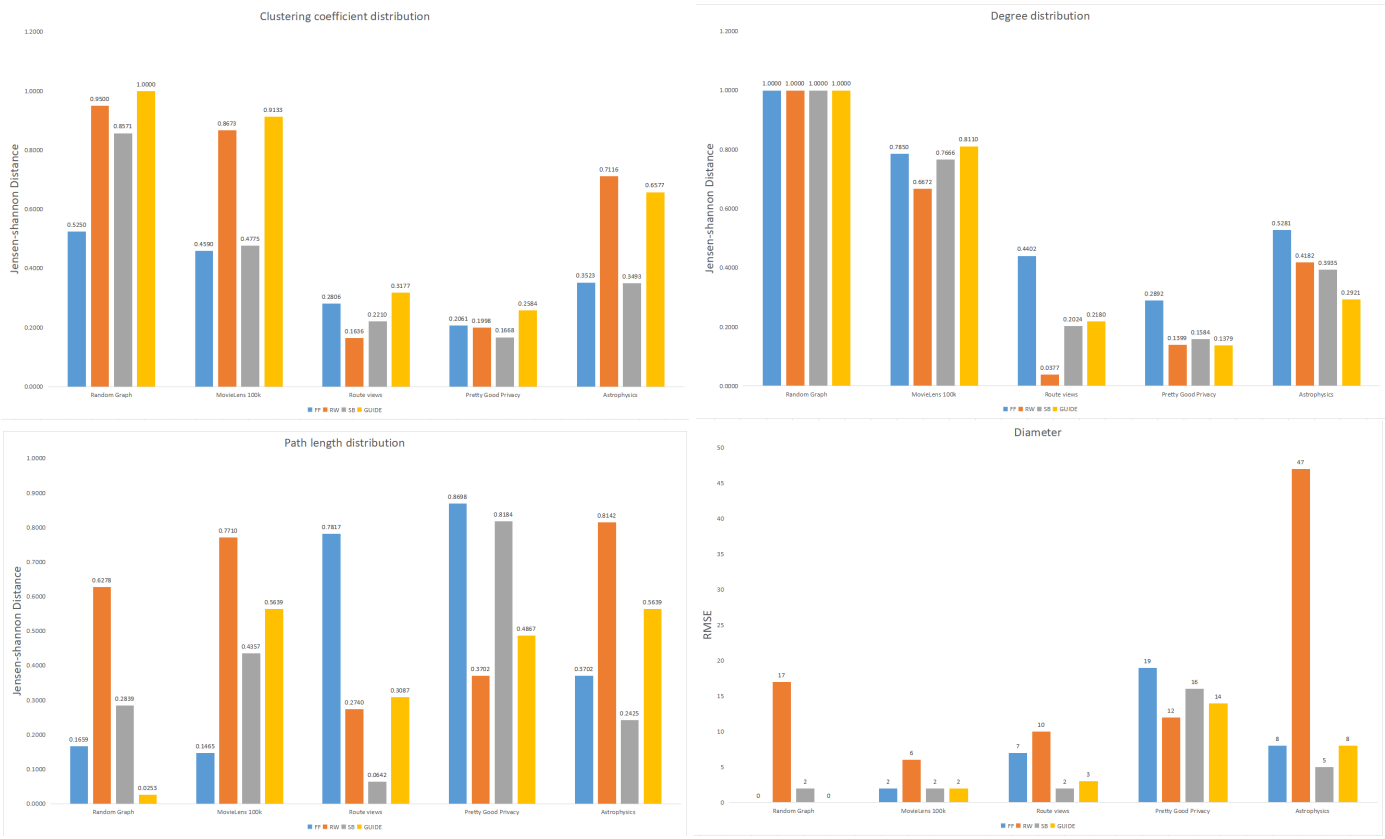

图 10: 与传统算法的效果比较

5 个图在 4 个不同抽样算法下的抽样子图与原图的比较结果如图 10所示。可以发现, 实际上 Guided Sampling 的效果没有特别显著的优异于其他算法, 但是效果基本上不会是这四种算法里面最差的。

黄色柱子是 Guided Sampling(GS), 第一幅图展示对聚类系数分布估计, 里面除了 Astrophysics 图 GS 比 SB 要优胜一些, 在其他 4 个图 GS 获得的子图与原图的分布差距都是最大的, 即效果是最差的。第二幅图在度分布估计里面,GS 效果总体也差强人意,Pretty Good Privacy 与 Astrophysics 里面效果好一点。第三幅图在路径分布的比较,GS 效果良好, 处于一个中规中矩的状态, 没有 RW 与 FF 那么差的效果, 与 SB 的抽样效果不相伯仲。第四幅图在直径的估计中,GS 也获得良好的效果, 基本上在这四种方法里面处于最好或者次好的水平。

通过四个属性的比较, 可以发现 GS 在本次实验与传统算法的比较并没有占特别大的优势, 在对于路径、直径此类全局属性比局部属性相对而言优于传统算法, 对于节点属性或者说是局部属性的捕获略差。

4.2.2 指导属性的准确性对抽样效果的影响

5 个图在平均度正确, 但平均聚类系数分别为正确值的 $\times 0.5, \times 0.75, \times 1.25, \times 1.75$ 倍时的抽样效果如图 11所示。

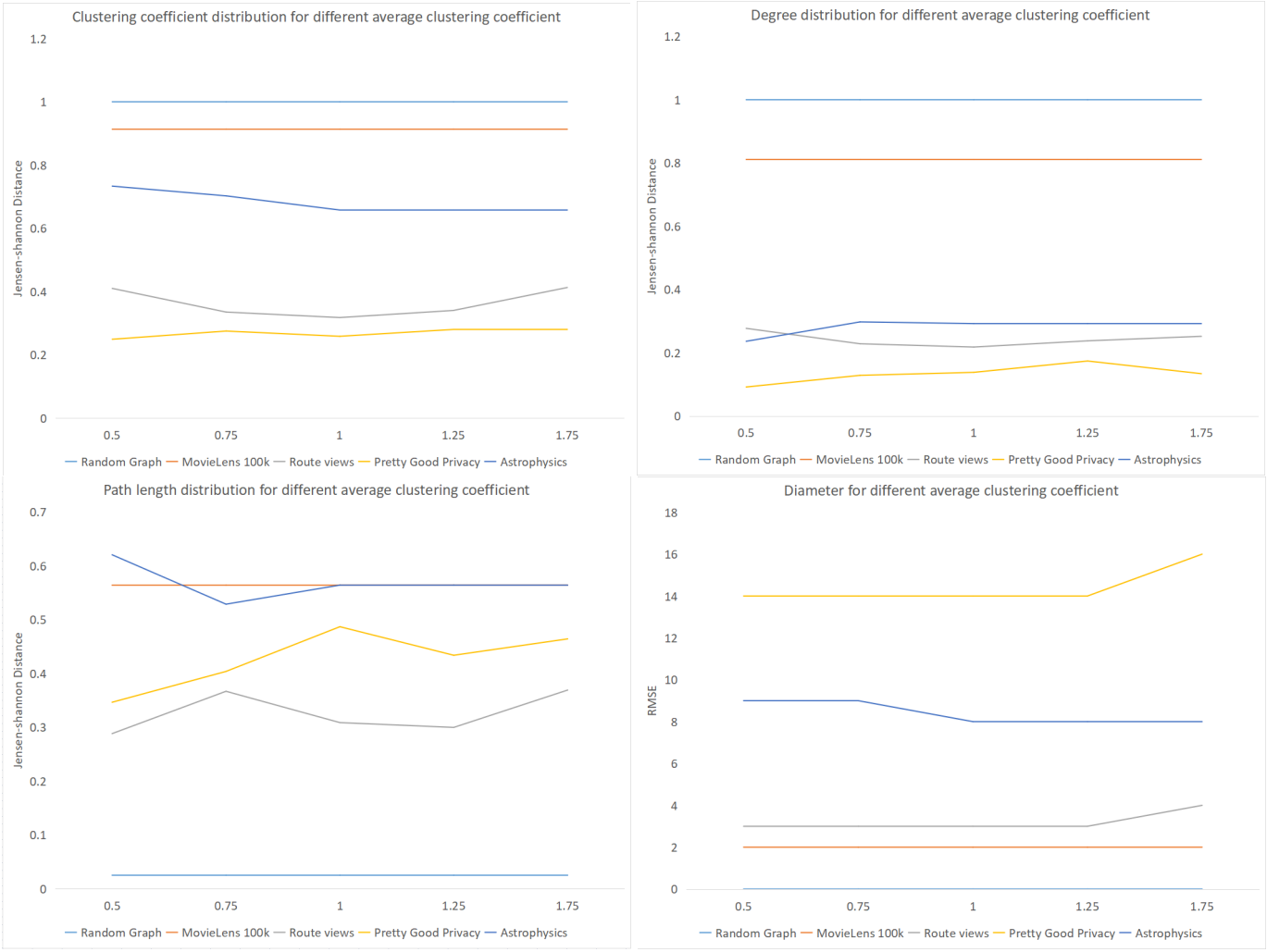

图 11: 不同平均聚类系数的效果比较

实验探究的是不同平均聚类系数输入对抽样结果的影响, 从图 11的左上图可以发现, 平均聚类系数估计对聚类系数分布的影响不大,Astrophysics 在平均聚类系数低于正确值的时候估计不准确, 在在平均聚类系数高于正确值的时候估计效果与准确时候一样, 而 Pretty Good Privacy 则反过来, 在平均聚类系数低于正确值的时候估计比输入正确值还好, 在在平均聚类系数高于正确值的时候估计效果与输入准确时候一样。Route views 的效果则与预想一样, 平均聚类系数输入越不准确估计也越不准确。从右上图展示平均聚类系数对度估计的影响, 在 Astrophysics 度分布的影响同聚类系数分布, 低于使结果变差, 高于不影响,Route views 则低于使结果变好, 高于不影响,Pretty Good Privacy 则低于使结果变好, 高于变差。平均聚类系数对路径分布和直径的估计影响明显, 而且没有明显规律性, 而且并非输入准确的平均聚类系数效果就更优。一个有趣的发现,MovieLens 100k 与 Random Graph 抽样子图的效果不随平均聚类系数输入的准确性影响。

5 个图在平均聚类系数正确, 但平均度分别为正确值的 $\times 0.5, \times 0.75, \times 1.25, \times 1.75$ 倍时的抽样效果如图 12所示。

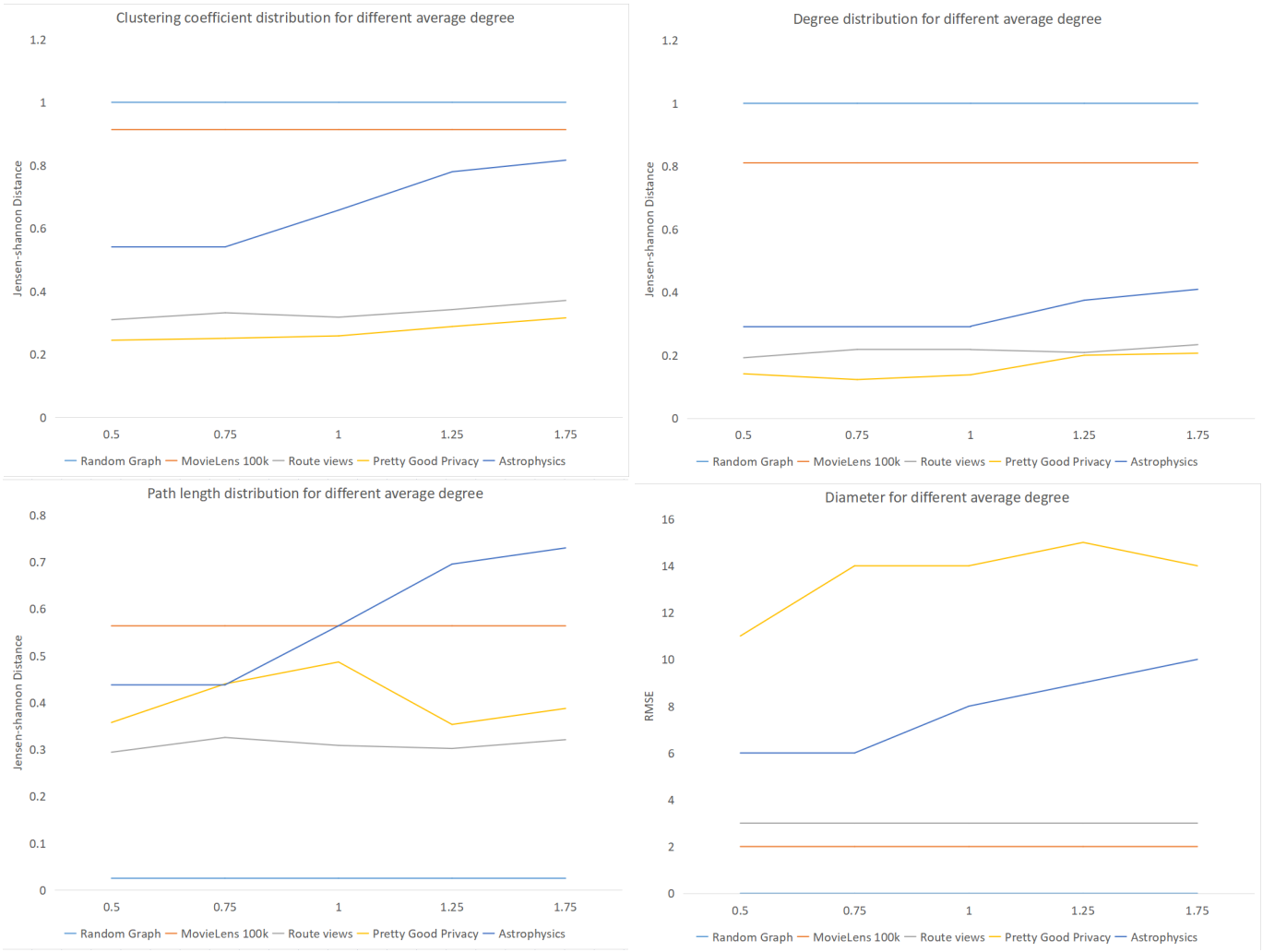

图 12: 不同平均度的效果比较

实验探究的是不同平均度输入对抽样结果的影响, 从图 12的左上图可以发现, 平均度对 Astrophysics 度分布、聚类系数分布、路径分布、直径估计的影响明显, 平均度低于使结果变优, 高于使结果变差。平均度对 Pretty Good Privacy 在聚类系数分布与度分布的估计的影响不明显, 对路径分布与直径的影响较为明显, 但是并没有明显规律。平均度对剩下图的属性估计影响效果不明显。

原论文也做了类似的探究实验, 得出来的结果也差不多, 单一研究平均聚类系数与平均度对抽样效果的影响时都没发现明显规律, 原论文认为是两属性具有相关性, 单一研究的时候体现不出影响效果, 本工作也承认这样的观点, 因此在原文基础上加了多组实验, 比较两指导属性不同组合形式下的影响如图 13所示。

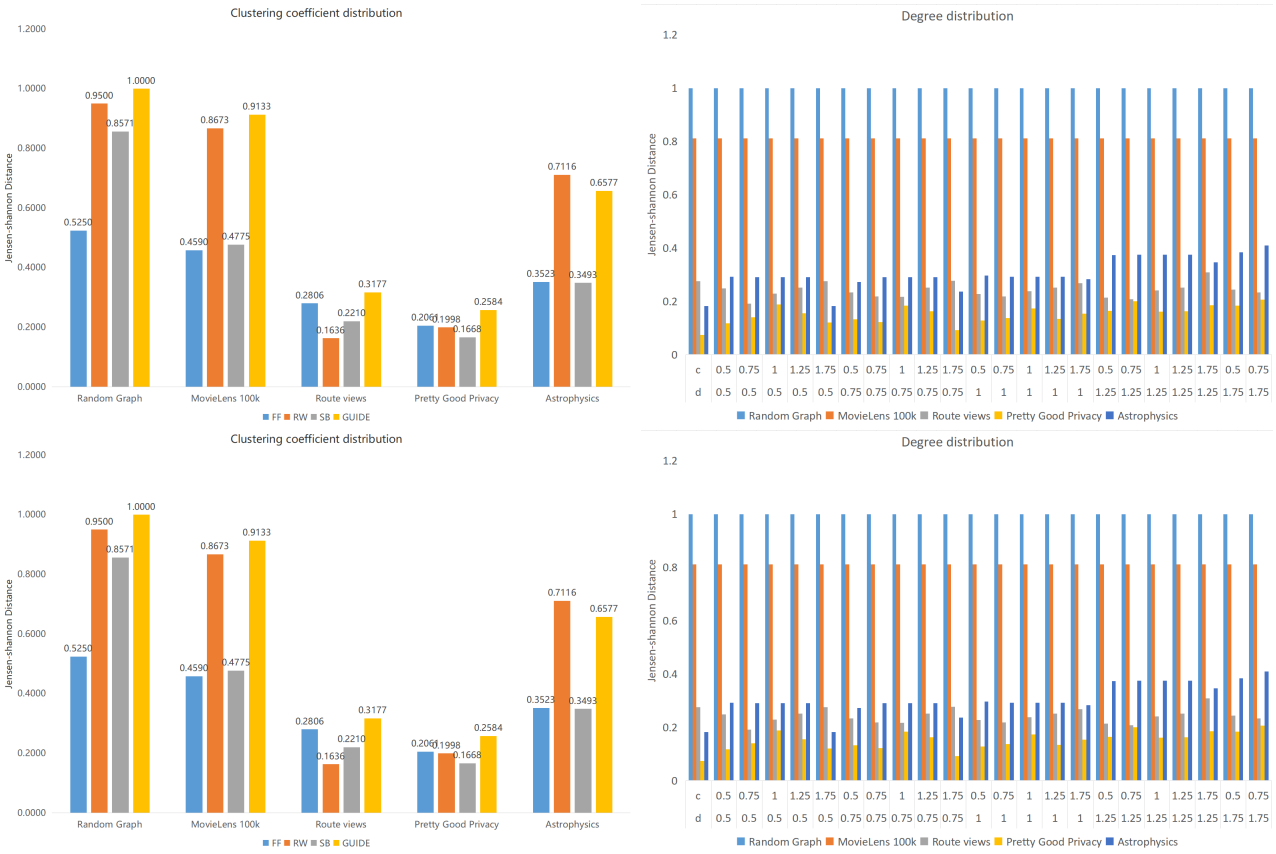

图 13: 不同两指导属性对不同属性估计的影响

从这四个属性的估计效果可以发现, 确实具有一定规律, 估计效果如锯齿状, 而且两指导属性都是准确的时候在四个估计属性上均不是最优组合, 当平均度输入低于真实值的时候, 平均聚类系数的对抽样的影响是与预期一样: 输入指导属性不准确, 抽样效果差, 输入指导属性越准确, 抽样效果越好; 但是在平均度输入高于真实值的时候, 抽样效果受指导抽样的影响就没有呈现规律性了, 而且四个估计属性较输入低平均度时效果都不好, 有理由猜测平均度对于指导抽样的过程起到关键作用。

5 总结

5.1 创新点

复现论文的创新点:

(1) 借助容易获得的先验知识, 将样本量减少到 1% 以下, 同时保持原始图的关键属性, 即度、聚类系数、路径长度、直径和分类性。

(2) 抽样方法利用节点的本地信息, 不需要任何全局信息。通过探索抽样节点的邻域来遍历图的一小部分, 这使得本论文的方法成为爬行在线网络的实用选择。

本工作创新点:

(1) 实现了本工作提出的指导图抽样方法 (Guided graph sampling)。

(2) 实现了论文提出的一种获得高平均度连通子图的图抽样算法 ModDFS。

(3) 模拟了如果指导属性欠抽样、过抽样、真实值的情况, 对指导属性在指导抽样的作用较原论文进一步探讨与了解。

5.2 总结与未来展望

本工作对论文[2]所提出的利用平均度与平均聚类系数来指导图抽样的算法 Guide sampling 进行复现。该算法没有开源代码, 因此本工作在 Python 与 R 两个平台都实现的算法复现。复现之后对实现的算法在真实图上进行实验验证, 并且与传统算法进行比较, 也对平均度与平均聚类系数对算法的影响进行了实验验证与探讨。

实验结果发现抽样子图的结果并不如论文所说的那么优越, 在 1% 的抽样比例下, 总体效果与传统算法差别不大, 但是 Guide sampling 还是有优势的, 在全局属性的估计中较传统算法具有优越性。实际上, 原文的实验结果也表明了 Guide sampling 与传统算法比较不具有绝对的优越性, 即并非在所有图的所有属性估计都优于传统算法, 只是在部分实验图的部分估计属性上有较好的效果, 但是总体估计效果是比较良好的, 因为 Guide sampling 不会是最差的估计效果, 综合效果理想。

对于平均度与平均聚类系数两指导属性对抽样结果的影响, 实验模拟了当输入指导属性不准确时候的指导抽样的效果。结果发现, 指导属性的细微偏差并没有非常明显的影响抽样过程, 而且影响的程度没有规律性, 论文里面认为是因为两属性具有相关性单一研究其中一个难以发现其中规律, 本工作也认同这一观点, 因此做了多组两指导属性不同情况下的组合实验, 发现当同时研究两属性时候是可以发现一些规律的, 可以合理认为两指导属性具有一定相关性, 而且实验发现最优的抽样效果并没有出现在当两指导属性都输入正确的时候, 至于准确的相关性如何, 可以扩大输入指导属性的偏差范围进一步实验。

实际上, 无论是原论文还是本工作的实验结果都表面了 Guide sampling 在抽样比例较小的情况下的大图抽样较传统算法具有一定优越性, 但是不多, 而且 Guide sampling 较其他算法的复杂度更高, 时间与空间资源消耗更大, 本文认为不具有现实推广的意义, 但是论文提出的利用先验知识进行指导图抽样的思维值得学习。

参考文献

[1] OUSSOUS A, BENJELLOUN F Z, LAHCEN A A, Big Data technologies: A survey. Journal of King Saud University-Computer and Information Sciences, 2018, 30(4): 431-448.

[2] YOUSUF M I KIM S. Guided sampling for large graphs. Data mining and knowledge discovery, 2020, 34(4): 905-948.

[3] LEI M, ZHANG X, YANG J, Efficiently Approximating Top-k Sequential Patterns in Transactional Graphs. IEEE Access, 2019, 7: 62817-62832.

[4] KRISHNAMURTHY V, FALOUTSOS M, CHROBAK M, Reducing large internet topologies for faster simulations / / International Conference on Research in Networking. 2005: 328-341.

[5] LESKOVEC J FALOUTSOS C. Sampling from large graphs / / Proceedings of the 12th ACM SIGKDD international conference on Knowledge discovery and data mining. 2006: 631-636.

[6] AHMED N, NEVILLE J, KOMPELLA R. Network sampling designs for relational classification / / Proceedings of the International AAAI Conference on Web and Social Media: vol. 6: 1. 2012: 383-386.

[7] KURANT M, BUTTS C T, MARKOPOULOU A. Graph size estimation. ArXiv preprint arXiv:1210.0460, 2012.

[8] LEE S H, KIM P J, JEONG H. Statistical properties of sampled networks. Physical review. E, Statistical, nonlinear, and soft matter physics, 2006, 73(1 Pt 2): 016102.

[9] REZVANIAN A, RAHMATI M, MEYBODI M R. Sampling from complex networks using distributed learning automata. Physica A: Statistical Mechanics and its Applications, 2014, 396: 224-234.

[10] HU P LAU W C. A survey and taxonomy of graph sampling. ArXiv preprint arXiv:1308.5865, 2013.

[11] ENDRES D SCHINDELIN J. A new metric for probability distributions. IEEE Transactions on Information Theory, 2003, 49(7): 1858-1860. DOI: 10.1109/TIT.2003.813506.

[12] SETHU H CHU X. A new algorithm for extracting a small representative subgraph from a very large graph. CoRR, 2012, abs/1207.4825. arXiv: 1207.4825. http://arxiv.org/abs/1207.4825.

[13] GJOKA M, KURANT M, BUTTS C T, Walking in Facebook: A Case Study of Unbiased Sampling of OSNs/ /INFOCOM 2010. 29th IEEE International Conference on Computer Communications, Joint Conference of the IEEE Computer and Communications Societies, 15-19 March 2010, San Diego, CA, USA. IEEE, 2010: 2498-2506. https://doi.org/10.1109/INFCOM.2010.5462078. DOI: 10.1109/INFCOM.2010.5462078.

[14] HARDIMAN S J KATZIR L. Estimating clustering coefficients and size of social networks via random walk/ /SCHWABE D, ALMEIDA V A F, GLASER H, 22nd International World Wide Web Conference, WWW '13, Rio de Janeiro, Brazil, May 13-17, 2013. International World Wide Web Conferences Steering Committee / ACM, 2013: 539-550. https://doi.org/10.1145/2488388.2488436. DOI: 10.1145/2488388.2488436.

[15] BARABASI A L OLTVAI Z N. Network biology: understanding the cell's functional organization. Nature reviews genetics, 2004, 5(2): 101-113.

[16] KATZIR L HARDIMAN S J. Estimating clustering coefficients and size of social networks via random walk. ACM Transactions on the Web (TWEB), 2015, 9(4): 1-20.

[17] KUNEGIS J. KONECT: The Koblenz Network Collection/ /WWW '13 Companion: Proceedings of the 22nd International Conference on World Wide Web. Rio de Janeiro, Brazil: Association for Computing Machinery, 2013: 1343–1350. https://doi.org/10.1145/2487788.2488173. DOI: 10.1145/2487788.2488173.