山东大学 计算机科学与技术 学院

可视化技术 课程实验报告

|  |  |  |  |
| --- | --- | --- | --- |
| 学号：201900130176 | 姓名： 李伟国 | | 班级： 智能 |
| 实验题目： 理解Force-directed layout | | | |
| 实验学时： | | 实验日期： 2021/10/25 | |
| 实验目的：  用D3跑一下 fore-directed layout | | | |
| 硬件环境：  处理器：AMD Ryzen 5 3600 6-Core Processor 3.60 GHz  Ram 16.0 GB | | | |
| 软件环境： | | | |
| 实验步骤与内容：  Force-directed layout：  需要计算一组点的集合中任意成对的点的之间的相互作用力，这经常被叫做 N-body 问题。在network visualization中，我们要使用类似与在星球之间计算天体之间的引力，去计算布局。在network中的每个节点扮演者相互排斥的带电粒子的角色，而链接连个节点的边扮演着将两个节点拉在一起的弹簧。这这就是force-directed layout。这里的force 可以看做是所有点之间两两节点的斥力和边对两两节点的拉力。  可以通过官网的例子来感受一下这种force-directed 的布局。        通过改变Force strength ，可以发现，force strength 越大 ，节点彼此之间的距离就会变得越来越小，layout 就会显得越紧凑。  计算 N-body force 最直接的方法就是考虑所有节点彼此之间的相互作用力，并且将每个节点的相互作用力的贡献都加起来。但是这种最初始的办法有 N2 的复杂度，这在实际情况中时很不利的，所以要进行改进。下面是改进算法。  Improvement：Barnes-Hut Approximation  核心思想是：通过用它们的质心替换一组远距离点来近似远程力。作为少量错误的交换，该方案显着加快了计算速度，复杂度为 n log n 而不是 n2。  该近似方法的中心是spatial index（空间索引），一个空间的map 可以帮助我们将一组节点建模成一个质点。在2位平面上，我们可以使用quadtree 这个数据结构，这个数据结构可以递归的将空间中的平面区域细分成4个等大的象限。如果实在3位的空间中，我们可以类比的使用octree 将一个立方体分成空间中的8个子立方体。  主要有3步：   1. 构建空间 索引（spatial index）（quadtree 或者是 octree） 2. 计算质心 3. 估计力（force）   Step 1：construct the Quadtree  以二维的数据点为例。当我们将第一个点插入到Quadtree，这个被添加到树的最高级别的树根中。  接下来去插入另一个节点，需要将空间进行一次细分来扩展这颗树。在每次后续插入时，可能会添加更细粒度的单元格，直到所有点都位于它们自己的单元格中。  通过网站给的交互式接口来体验一下这个过程。  **刚开始，原始的输入图像(一些2维的输入点)**    添加一个节点到quadtree 中 添加第二个节点到quadtree中    此时可以发现，当添加第一个节点后，给节点就位于树的根部(外部最大的矩形框)，当再次添加第二个节点的时候，对空间进行细分，直到该节点单独的位于一个cell之中。  Step 2：calculate Center of Mass  构建好quadtree 后，就可以对树的每个cell 计算质心了。计算的方法很简单，每一个树的cell 的质心就是其四个孩子的中心的加权平均。首先访问quadtree的叶子cell，然后再访问其后续的父亲cell。合并那些我们向根节点传递个数据。一旦访问（遍历）全部节点，树的每一个cell都已经使用位置和它的质心更新过了。  Step 3：Estimate N-Body Forces  当计算给定点所受的力的时候，通过交互式的web接口，我们可以很直观的看清楚背会的计算原理。  如图，白色点是给定的点，要计算它和所有点之间的斥力的合力。（这个给定的点类似于高中物理所学的试探电荷，不妨在这里将其称作试探point）    通过移动给定点，可以发现，在不同的位置，其所受的其它节点对其斥力的和力是不一样的。  最开始的方法是：先忽略这个quadtree 这个结构，通过将所有的节点对给定的点所造成的斥力用向量加法计算其合力。  但是我们会使用quadtree去代替这个最原始的方法去加速计算和近似的远程的力。不是计算每个节点之间的相互作用力，我们可以计算给定点和质心的作用力就可以了，使用小的quadtree 的cell 去计算邻近的点，用大的cell去计算远的点的相互作用力。  但是有一个至关重要的点，那就是什么是“long-range” ，什么是“short-range” 的力？我们既要考虑到树的cell的距离，又要考虑cell的边长。如果比值 width/distance 小于我们设定的threshold（阈值），我们就认为是“long-range” forces并且使用该cell的质心。反之，我们将会在树中递归的访问每个子cell。  当threshold设置为1的时候，quadtree树的cell的质心会被使用。并且它的内部节点会被忽略，如果样本点的距离到cell的中心的距离大于或者等于cell的边长（即小于一定的阈值）。    此时样本点到cell的中心的距离要大于cell的边长，此时就将整个cell作为整体来表示对sample点的影响  当我们移动了prob的位置，可以发现，在离它最近的位置的点，prob到其cell的距离小于cell的边长，这是就递归的访问所有的子cell，而在prob的右边的cell中，prob到cell中心的距离大于cell的边长，此时就将整个Cell 作为整体访问（很大的红色的点）。  但我们不断的调整theta的值，我们就会发现越来越多的离根节点越近的CELL的中心到sample 点的句距离的比值会轻易的大于该阈值，那么就会有越来越多的大的CELL被当成整体被一次性访问，而不是递归的一个一个访问每个孩子cell。      θ等于0是一种比较极端的情况，即每一个cell的边长和sample point 到cell的距离的比值都会大于0，即都不是long-range force，所有要递归的访问每一个cell的每一个子cell。现在我们改变theta 的值，观察，有哪些cell 被判定成了long-range force，这些cell 就被当成整体，用其质点计算和sample point 之间的距离。  图表 1：theta = 0.0    当我们将theta的值设置的大一点的时候，存在一些cell，这些cell 的边长与其到smaple point的距离的比值小于该theta（阈值），这些cell将会被当做整体，其质点将被访问。  而哪些大于阈值的cell任然递归的访问所有的子cell。  图表 2：θ = 0.50  接下来持续的调大theta的值    图表 4 θ = 1.3  图表 3θ = 0.7  通过将theta 的值逐渐的调大，我们发现，又来越多的cell的边长和其到sample points的比值小于该阈值，那么这些cell对sample point的作用力都将被认为是long-range force，所以这些cell将被当做整体，用其质心只计算一次其对sample point 的作用力。  现在我们可以使用Barnes-Hut估计去限制总的比较的次数，并且该方法能够非常有效的加速。并且，当我们将theta 的值设置的越大，那么速度就越快。并且一个较好的theta的取值应该是1.  用d3 构建 directed-force tree：  用7 个节点来表示没一周的七个不同的天，然后拖动节点    可以发现力导向图的布局很好 | | | |
| 结论分析与体会： | | | |

附录：程序源代码