

HARP: Hierarchical Representation Learning for Networks

HAOCHEN CHEN, BRYAN PEROZZI*, YIFAN HU, STEVEN SKIENA

@AAAI'2018

*DEEPWALK 作者

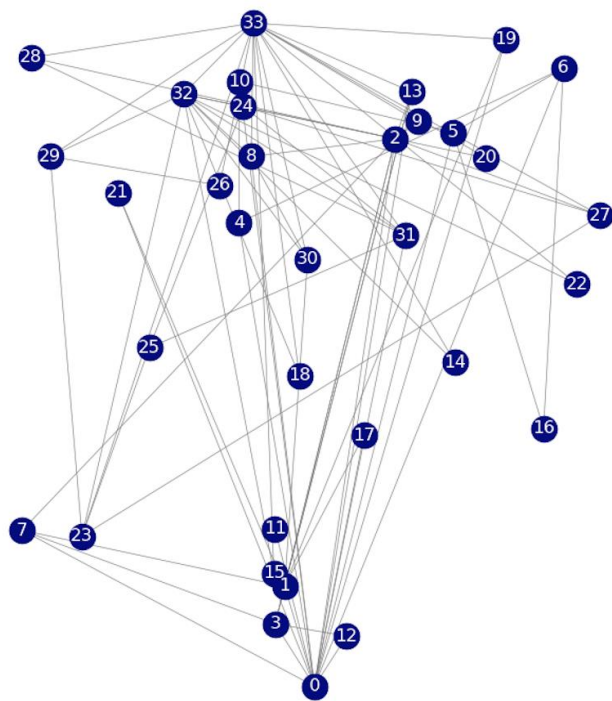
Motivation

- 现存的Graph Embedding算法都是迭代算法；
- 迭代算法收敛的结果与初始值的设置密切相关；
- 如果初始化不好，训练时就可能会陷入局部最优，训练出来的向量效果很差；
- 另外，现存的算法不能很好地捕捉远距离节点之间的关系。

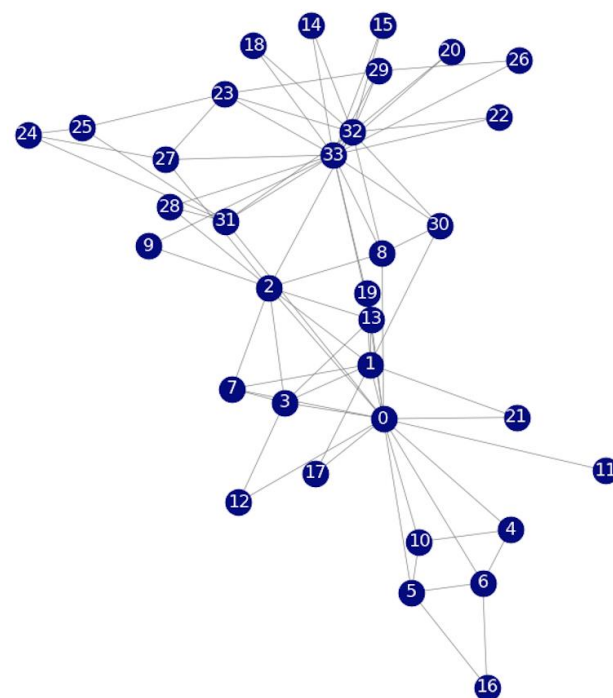
本文尝试解决这两个问题。

从Graph Drawing说起

Graph Drawing: 把一个图画出来。



节点在二维平面上随机分布
不能直观看出图的结构信息

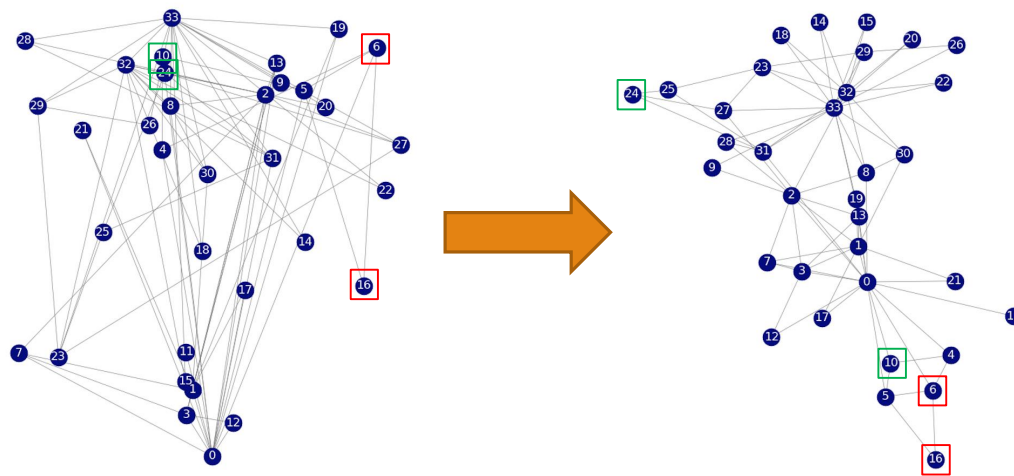


Graph Drawing算法后
可直观看出整个图存在4~5个社区

Graph Drawing

Graph Drawing可以看成是一个Graph Embedding任务，它的目标是：

- 在原图中距离较近的节点，画出来之后距离也要比较靠近；
- 在原图中距离较远的节点，画出来之后距离不应该很靠近；
- 节点间要有一定距离，每个节点都能看清。



Graph Drawing

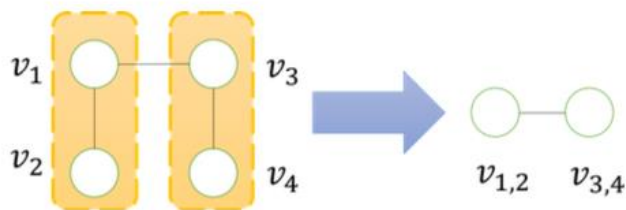
Graph Drawing基本思路:

- 合并距离较近的节点为一个簇;
- 为每个簇安排一个合适的初始位置;
- 展开每一个簇, 簇中的每个节点位置画在簇中心的附近。

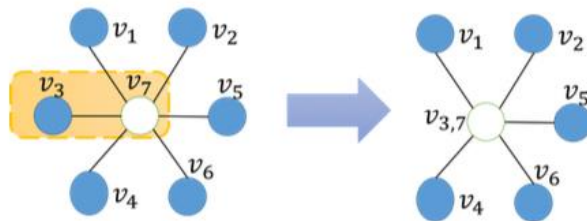
Graph Drawing

图的节点合并方法

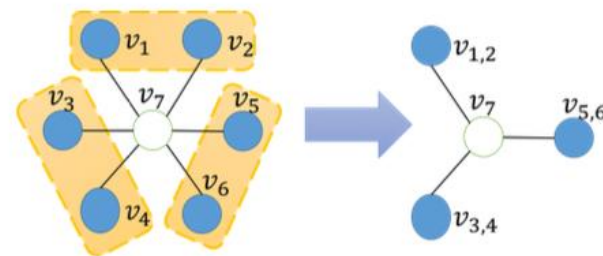
- 边坍缩：随机选择一些边，与一个节点的两条边不能同时被选择，合并这些选中的边；
- 星形坍缩：对于星形拓扑，随机选择外围节点合并。



(a) Edge Collapsing.



(b) Edge Collapsing fails to collapse stars.

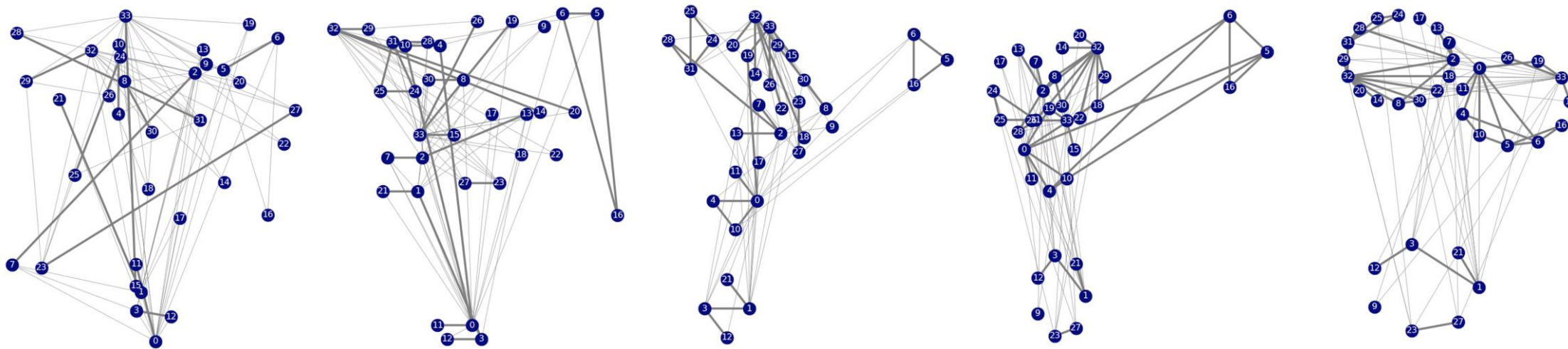


(c) Star Collapsing.

Graph Drawing

递归图合并

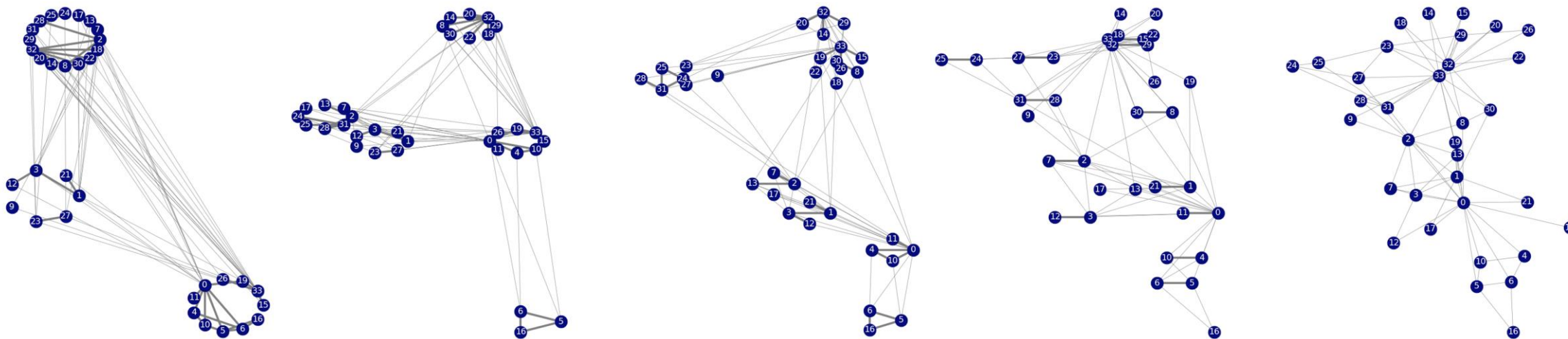
- 合并距离较近的节点为一个簇，产生新的图；
- 对产生的新图继续合并节点，直到节点数量小于一个阈值。



Graph Drawing

递归图展开

- 为每个簇安排一个合适的初始位置；
- 展开每一个簇，簇中的每个节点位置画在簇中心的附近；
- 递归展开中间合并的每一个图，得到最终结果。



Graph Drawing

Graph Drawing基本思路:

- 合并距离较近的节点为一个簇，产生新的图；
- 对产生的新图继续合并节点，直到节点数量小于一个阈值；
- 为每个簇安排一个合适的初始位置；
- 展开每一个簇，簇中的每个节点位置画在簇中心的附近；
- 递归展开中间合并的每一个图，得到最终结果。

Hierarchical Representation Learning

- 使用图合并算法对原始图进行递归合并，产生多个层级的图；
- 从最后的图开始训练，得到每个簇节点的特征向量；
- 使用簇节点的特征向量作为这个簇里所有节点的向量的初始值，训练下一个层级的图的节点特征向量；
- 对每个层级的图进行递归训练，直到得到最后每个原始节点的特征向量。