Author = Song Jian

阅读论文：Inductive Representation Learning on Large Graphs

|  |  |  |
| --- | --- | --- |
| @article{Hamilton2017,  author = {Hamilton, William L},  number = {Nips},  pages = {1--19},  title = {{Inductive Representation Learning on Large Graphs}},  year = {2017}  } | [1] W. L. Hamilton, “Inductive Representation Learning on Large Graphs,” no. Nips, pp. 1–19, 2017.  源码：<https://github.com/williamleif/GraphSAGE>  笔记有参考，知乎用户：浅梦，蝈蝈 | 斯坦福大学 |

#### 文档由以下几个部分组成：

* 文章简述
* 算法原理
* 论文实验
* 代码部分

1文章简述

统计机器学习可以分成两种: transductive learning, inductive learning。

* transductive(直推式)：To specific (test) cases, 指的是测试集是特定的(固定的样本);
* Inductive(归纳式): 测试集不是特定的，会继续添加新的节点。

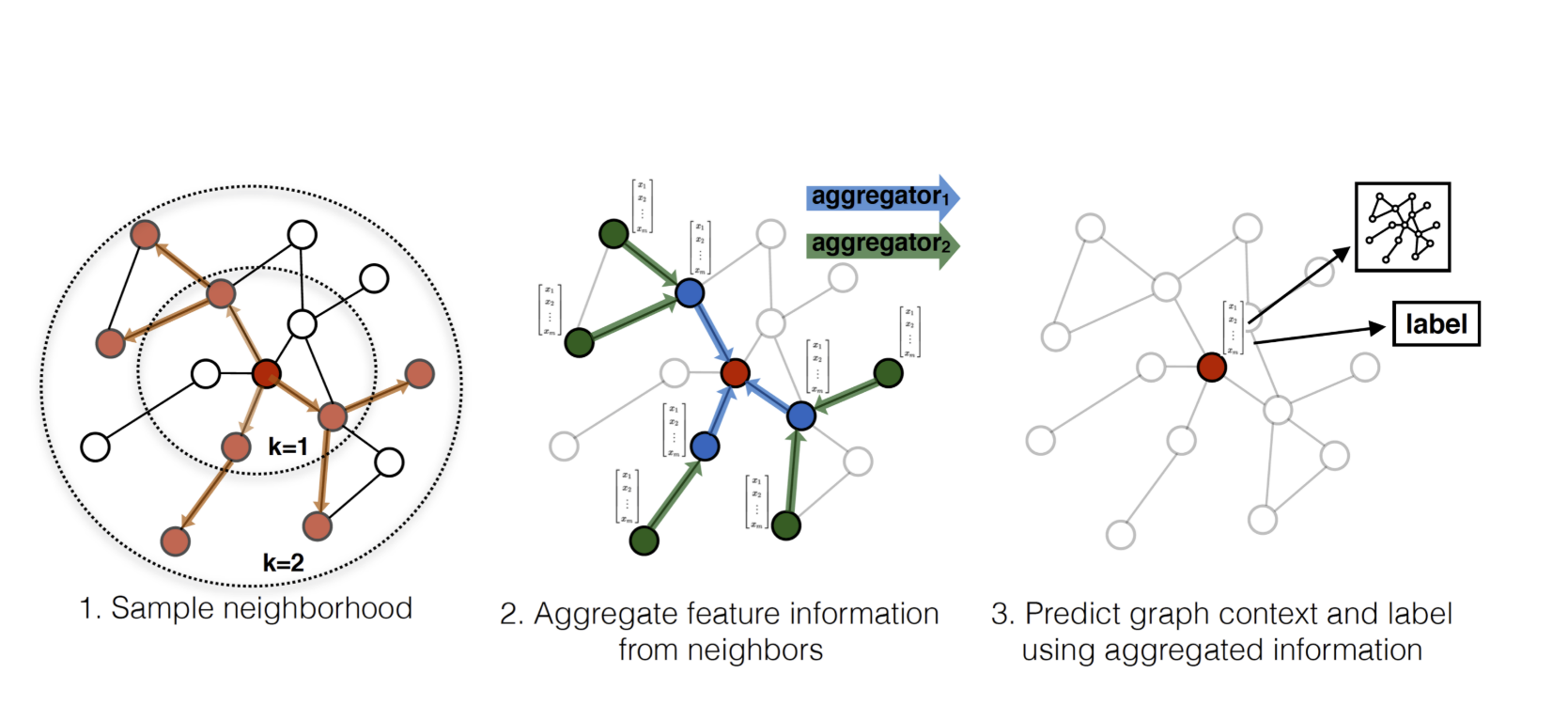
GNN中经典的DeepWalk, GCN方法都是transductive learning。

GCN算法的输入从固定位置和大小的像素图变成了一个表达网络间各个点的邻接矩阵和一个对应点的特征的矩阵的拼接，有了格式化的矩阵，我们可以把他们映射到神经元层，一层不行两层至n层，最后接一个softmax，总是可以得到一个比较理想的基于点的分类结果。但是这样的方法有缺陷，由于输入的矩阵大小是固定的，因此增加节点，模型就需要重新训练。

斯坦福Jure教授组提出一种适用于大规模网络的归纳式（inductive）学习方法-GraphSAGE，是一个inductive的框架，可以利用顶点特征信息（比如文本属性）来高效地为没有见过的顶点生成embedding。

与GCN不同的是，GraphSAGE不是对所有的邻居进行聚合，而是对局部邻居采样并聚合特征，其核心思想是通过学习一个对邻居顶点进行聚合表示的函数来产生目标顶点的embedding向量。

2 算法原理



论文原图 1

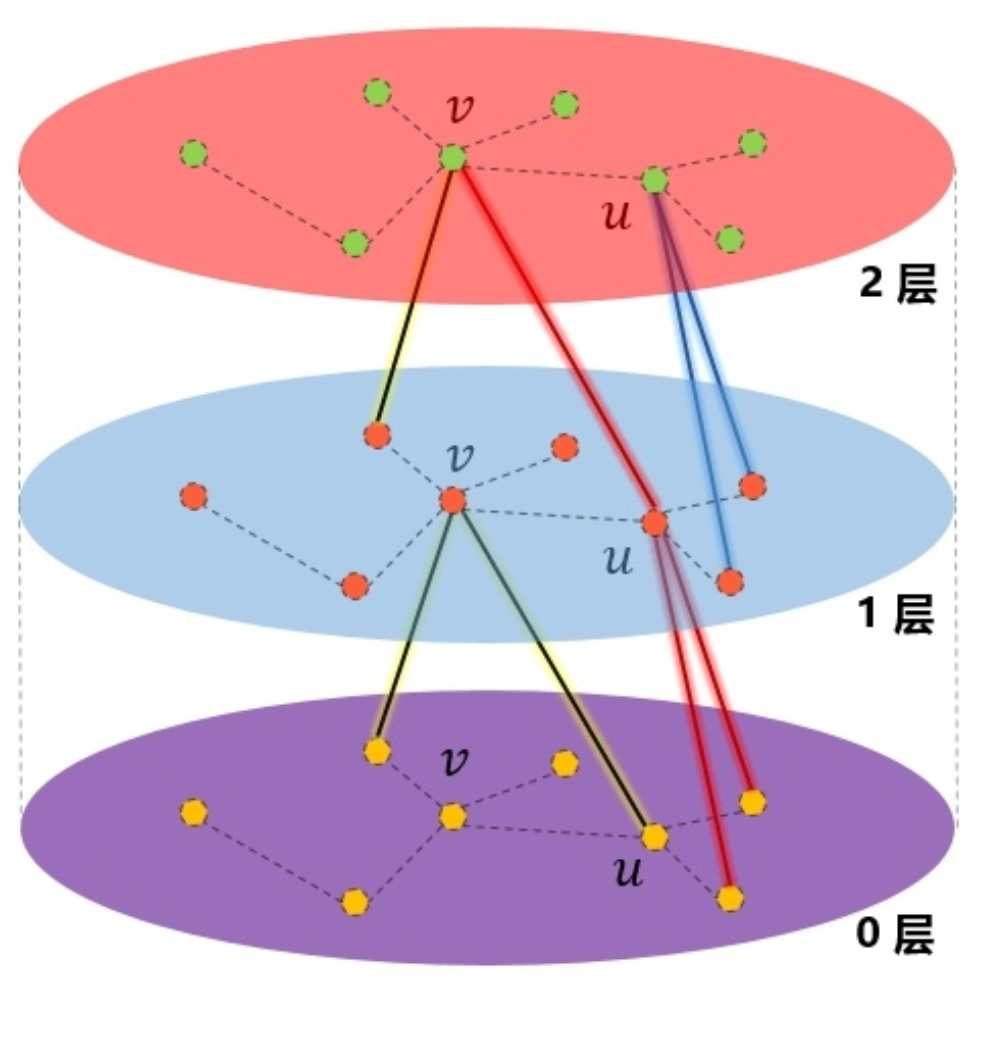
GraphSAGE 是Graph SAmple and aggreGatE的缩写，其运行流程如上图所示，可以分为三个步骤：

1. 对图中顶点邻居顶点进行采样

2. 根据聚合函数聚合邻居顶点蕴含的信息

3. 得到图中各顶点的向量表示供下游任务使用

出于对计算效率的考虑，每次采样一定数量的顶点邻居作为，比如K个，如果当前顶点的邻居不足K个，则使用有放回的采样方法，如果邻居数量大于K个，则采用无放回的采样。



引自知乎用户：蝈蝈

2.1 聚合函数(aggregator function)的选择

这篇文章尝试了多种aggregator function。aggregator 的作用是把一个向量的集合转换成向量,也就是聚合。和其他机器学习任务中的数据（如图像，文本等）不同，图中的节点是没有顺序的。aggregator function操作的是一个无序的向量集合 。所以这个aggregator function需要symmetric,或者说 invariant to permutations of its input。就是输入顺序不影响函数结果。作者提供了三种方式：

* Mean aggregator

显然对向量集合，对应元素取均值是最直接的想法。

这篇文章说取均值和图卷积是等价的，推导出了一种GCN方法的变体。

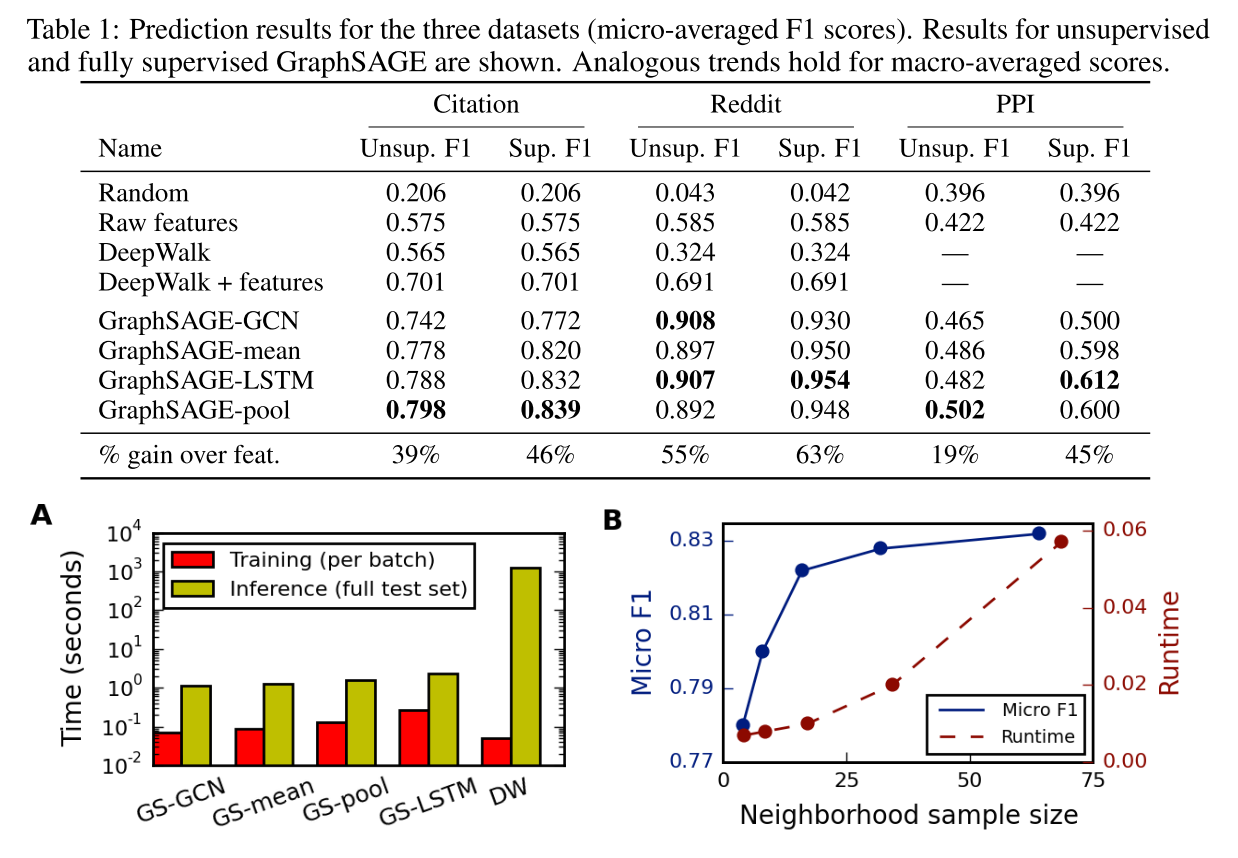
* LSTM aggregator

和mean aggregator相比，LSTM有更大的表达能力。但是LSTM不符合symmetric的性质，输入是有顺序的。所以把相邻节点的向量集合随机打乱顺序，然后作为LSTM的输入。

* Pooling aggregator

尝试了pooling做aggregator, 所有相邻节点的向量共享权重，先经过一个非线性全连接层，然后做max-pooling。

3论文实验

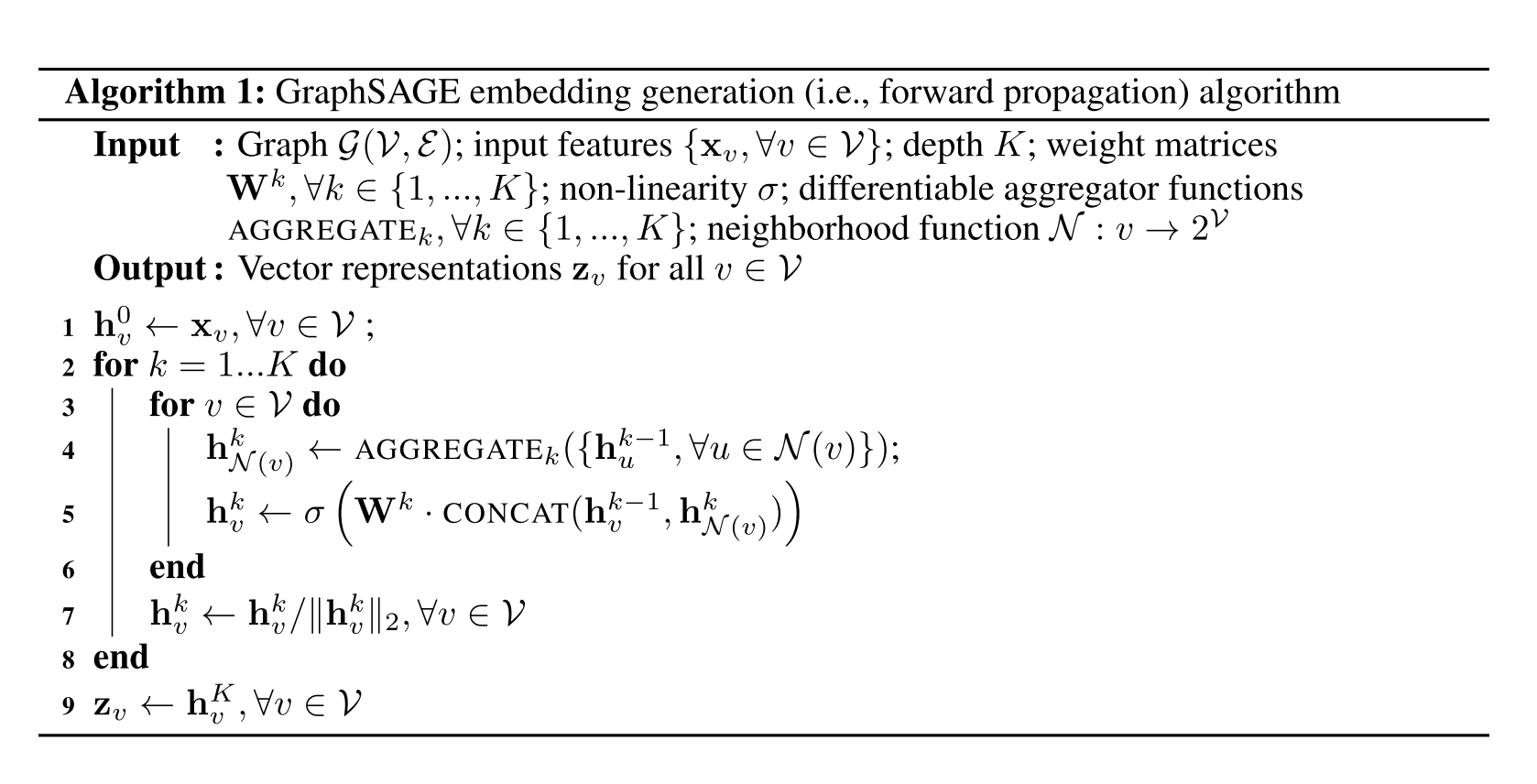


论文实验 1

数据集：Citation，Reddit，PPI

可以看出有监督比无监督效果好，一般是LSTM或Pooling效果比较好

4代码部分



原文代码 1

第2-7行：两个循环，外层循环表示K，即深度，内层 循环表示每个节点𝑣∈𝒱

第4行：表示节点v的任意相邻节点 的聚合信息的集合。 是一个向量，表示从相邻节点处获得的信息。AGGREGATE表示一个可微分的聚合函数

第5行：表示获取到的信息与节点自身的信息进行拼接