

- 阅读论文Fast Network Embedding Enhancement via High Order Proximity Approximation

- Task Definition

- 动机

- 论文对于当前的大部分网络表示学习的方法进行了研究和统计，从一种高屋建瓴的角度来看待现有的网络表示学习的方法，并将这些方法总结成一套通用的框架，并论证了通过构建高阶邻近矩阵可以改善网络表示学习的效果

- 主要模型：

通用的框架分为以下两个步骤：

- 邻近矩阵的构建
      - 降维

- K阶邻近矩阵的定义

归一化的邻接矩阵和拉普拉斯矩阵构建了一阶邻近矩阵，一阶邻近矩阵模拟了顶点间的近似情况。但是现实世界中的网络往往是十分稀疏的，所以一阶邻近矩阵不足以模拟顶点间的相似情况，所以，学者通常扩展高阶邻近矩阵去模拟顶点间的相似情况。例如，二阶邻近矩阵表示两个顶点含有共同的邻居，也代表着两个顶点之间可以通过2步游走到达。本文以此来拓展高阶的情况，K阶邻近矩阵定义为通过K步游走可到达。定义公式如下所示：

$$A^k = A \cdot A \dots A$$

- 网络表示学习统一框架

网络表示学习的框架分为以下两个部分：邻近矩阵的构建，降维

- 邻近矩阵的构建

计算邻近矩阵  $M \in R^{|V| \times |V|}$ ，其中  $M$  是K阶邻近矩阵的多项式形式，例如

$$M = \frac{1}{k} A + \frac{1}{k} A^2 + \frac{1}{k} A^3 \dots + \frac{1}{k} A^k$$

- 降维

记网络嵌入矩阵  $R \in R^{|V| \times d}$  和上下文嵌入矩阵  $C \in R^{|V| \times d}$  所以  $R \cdot C^T$  应当逼近邻近矩阵  $M$  不同的算法可能需要从对于  $M$  和  $R \cdot C^T$  的距离计算的方式也不一样。

- deepwalk示例

文中还将Spectral Clustering,DeepWalk,GraRep三种方法放入统一的框架中。因为对于另外两种方法不熟悉，所以这边就只展示DeepWalk这种方法在这种框架下的剖析。

DeepWalk对于每个顶点学习两个表示，分别是顶点本身的表示和顶点作为上下文时的表示，记作  $R \in R^{|V| \times d}$  和  $C \in R^{|V| \times d}$ 。DeepWalk实质是将矩阵  $M$  分解为  $R \cdot C^T$ 。其中  $M = \log \frac{A + A^2 + \dots + A^w}{w}$

- network embedding改进

作者通过对三种network embedding方法的观察，给出了以下观察结果

- 通过构建更高阶和更准确的邻近矩阵能够获得更好的网络表示效果。
      - 准确的高阶邻近网络的计算量对于大规模网络是不可行的

第一点驱动我们去获得更高阶的邻近矩阵，但是第二点又要求我们不能去获得高阶的邻近矩阵。所以作者的想法是充分利用现有的临近矩阵。

本文目标是学习到更好的节点表示 $R'$ 和更好的上下文表示 $C'$ 去拟合 $g(A)$ ,  $g(A)$ 是相比于 $f(A)$ 即M更高阶的邻近矩阵

## ■ 模型

本文提出 $R' = R + \lambda A \cdot R$ 并且 $C' = C + \lambda A^T \cdot C$ , 其中

上述两式相比于原先的 $R$ 和 $C$ 吸收了更高阶的信息, 把两式相乘, 得到式子:

$g(A) = f(A) + 2\lambda A f(A) + \lambda^2 A^2 f(A)$ ,  $f(A)$ 是K阶邻近矩阵多项式,  $g(A)$ 是K+2阶邻近矩阵多项式。

还可以将公式泛化成如下形式:

$$R' = R + \lambda_1 A \cdot R + \lambda_2 A \cdot (A \cdot R),$$

$$C' = C + \lambda_1 A^T \cdot C + \lambda_2 A^T \cdot (A^T \cdot C)$$

## ■ 实验结果

% Labeled Nodes	% Accuracy			Time (s)
	10%	50%	90%	
GF	50.8 ( <b>68.0</b> )	61.8 ( <b>77.0</b> )	64.8 ( <b>77.2</b> )	4 (+0.1)
SC	55.9 ( <b>68.7</b> )	70.8 ( <b>79.2</b> )	72.7 ( <b>80.0</b> )	1 (+0.1)
DeepWalk <sub>low</sub>	71.3 (76.2)	76.9 (81.6)	78.7 (81.9)	31 (+0.1)
DeepWalk <sub>mid</sub>	68.9 ( <b>76.7</b> )	76.3 (82.0)	78.8 (84.3)	69 (+0.1)
DeepWalk <sub>high</sub>	68.4 ( <b>76.1</b> )	74.7 (80.5)	75.4 (81.6)	223 (+0.1)
LINE <sub>1st</sub>	64.8 (70.1)	76.1 (80.9)	78.9 (82.2)	62 (+0.1)
LINE <sub>2nd</sub>	63.3 ( <b>73.3</b> )	73.4 (80.1)	75.6 (80.3)	67 (+0.1)
node2vec	76.9 (77.5)	81.0 (81.6)	81.4 (81.9)	56 (+0.1)
TADW	78.1 (84.4)	83.1 (86.6)	82.4 (87.7)	2 (+0.1)
GraRep	70.8 (76.9)	78.9 (82.8)	81.8 (84.0)	67 (+0.3)

上图是节点分类的结果图, 括号内的是利用本文方法加强效果之后的结果和相应增加的时间, 可以看出时间的增加微乎其微, 证明了作者以几乎不增长时间的方式, 改善了网络表示学习的效果。