【Node2vec】基于网络的表示学习和特征提取

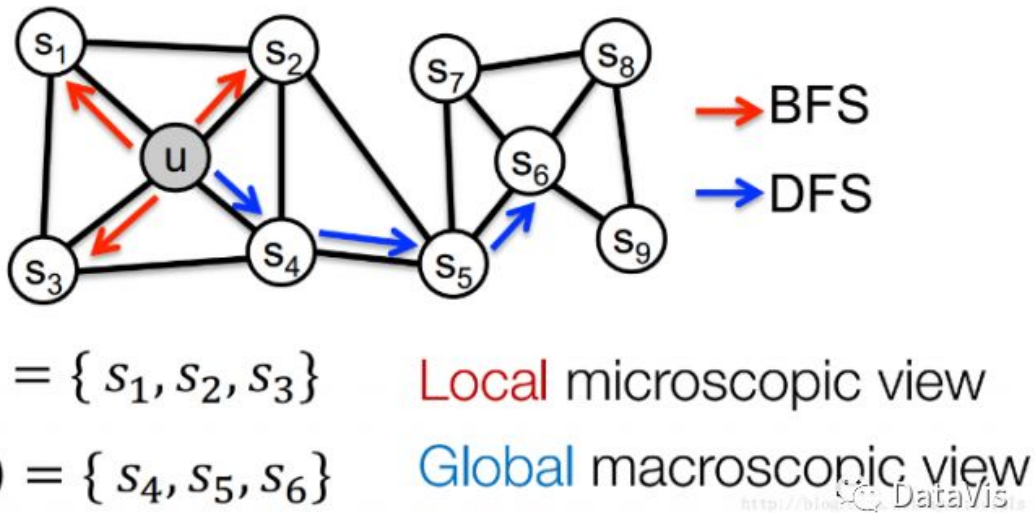
ZUFEVIS DataVis 2018-08-09

Node2vec: 基于网络的表示学习和特征提取

概述

该文指出，在网络分析中对节点和边的预测是是十分重要的工作，如节点分类和链路预测。在典型的节点分类中，人们会对网络节点最有可能的的标签感兴趣，如社交网络中的用户兴趣标签，蛋白质网络中的功能标签等。链路预测则可以帮助人们预测和发现节点间新的相互作用。

然而，实际网络中的结构特征是复杂多样的。特别地，网络中的节点可以基于homophily或structural equivalence等特征进行分类，也即是具有两种常用的相似性度量的方法——内容相似性和结构相似性。如下图所示，其中内容相似性是指相邻节点之间的相似性（u和S_1），而结构相似性是指结构上相似的点，其距离不一定相近（u和S_6）.



因此，该文提出一种有效结合了random walk和Word2vec的用于网络中特征学习的可扩展算法node2vec。其中，2nd-order random walks受deepwalk启发，同时考虑BFS和DFS两个方面，并通过设定参数控制并实现对节点进行采样。而Word2vec模型则可以在高效表征网络节点特征的同时有效保留节点的上下文联系。为了进一步验证该算法的有效性，作者们在社交网络、信息网络等多个实际网络上对多标签分类、链路预测、计算性能等多方面进行了性能评估。通过与现有的特征学习算法的对比分析，有效证明该文算法具有良好的表现和竞争性。

实现步骤

一、将特征抽取转换成一个最优化目标函数的问题

1. 定义目标函数以描述目标节点能保存相邻节点的可能性

$$\max_f \sum_{u \in V} \log Pr(N_S(u)|f(u)).$$

其中，$N_{S(u)}$表示通过采样策略S得到的节点u网络邻域，f是大小为|V|×d的矩阵。

2. 定义所需条件

①条件独立性

$$Pr(N_S(u)|f(u)) = \prod_{n_i \in N_S(u)} Pr(n_i|f(u)).$$

②节点之间的对称性

$$Pr(n_i|f(u)) = \frac{\exp(f(n_i) \cdot f(u))}{\sum_{v \in V} \exp(f(v) \cdot f(u))}$$

3. 代入得到最终的目标函数

$$\max_f \sum_{u \in V} \left[-\log Z_u + \sum_{n_i \in N_S(u)} f(n_i) \cdot f(u) \right]$$

其中，

$$Z_u = \sum_{v \in V} \exp(f(u) \cdot f(v)).$$

二、设计节点采样策略——随机游走

记开始节点为c_0=u，c_i为随机序列步中的第i个节点，随机游走选择下一个节点的公式为：

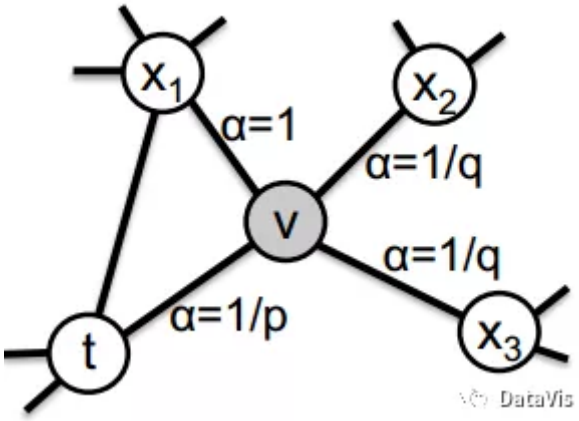
$$P(c_i = x \mid c_{i-1} = v) = \begin{cases} \frac{\pi_{vx}}{Z} & \text{if } (v,x) \in E \\ 0 & \text{otherwise} \end{cases}$$

即若图E存在边(v,x)，则以概率(π_{vx})/Z选择下一节点x，其中π_{vx}是非正则化的v到x的转移概率，Z是正则化常数。其中最简单的情况是令边的权重π_{vx}=w_{vx}。

而该文的2nd-order random walks将π_{vx}=w_{vx}，改进为π=α_{pq}(t,x)*w_{vx}

$$\alpha_{pq}(t,x) = \begin{cases} \frac{1}{p} & \text{if } d_{tx} = 0 \\ 1 & \text{if } d_{tx} = 1 \\ \frac{1}{q} & \text{if } d_{tx} = 2 \end{cases}$$

其中，d_{tx}表示节点t和x之间的最短路径距离，p控制返回walk中已访问节点的概率，当p>max(q,1)时，对已访问节点采样的概率较低，当p<min(q,1)时，反之；q则可以区分搜索方向，当q>1时，随机游走偏向于接近节点t的节点，近似于BFS，当q<1时，则反之，更近似于DFS。



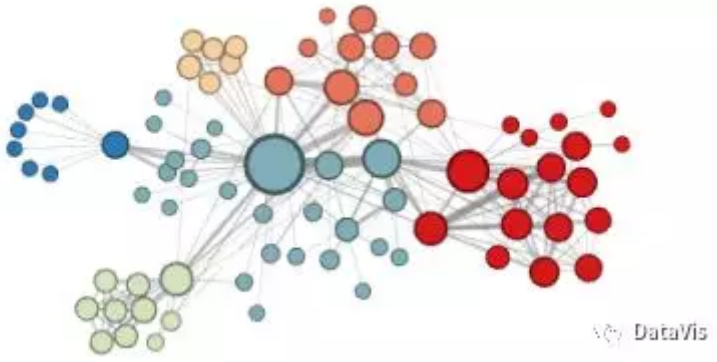
▲如上图，当前一节点t和下一节点x与当前节点v距离相等时，α=1,；当下一节点x为上一节点时，也即是跳转回t时，α=1/p，其余情况为α=1/q。

因此，本文是将随机游走生成的多个节点序列当做文本输入，使用word2vec中的skip-gram模型生成向量，并利用负采样方式减少计算开销。

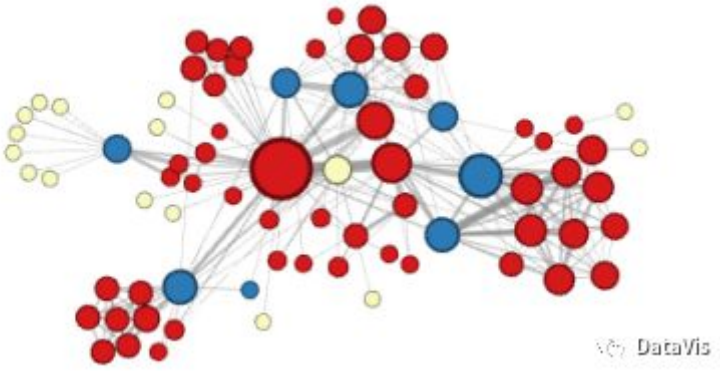
案例分析

(1) 参数验证

首先在Les Misérables network（小说悲惨世界的字符网络）中，通过调节不同的p，q验证node2vec可以遵循并实现BDS和DFS策略的原则。



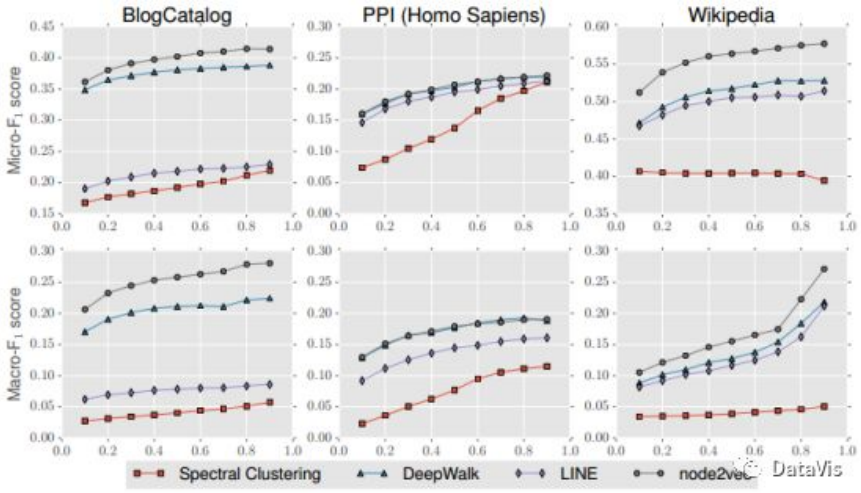
▲表示p=1，q=0.5时，基于homophily的反映结果



▲表示的是p=1，q=2时，基于structural equivalence的反映结果

(2) 多标签分类

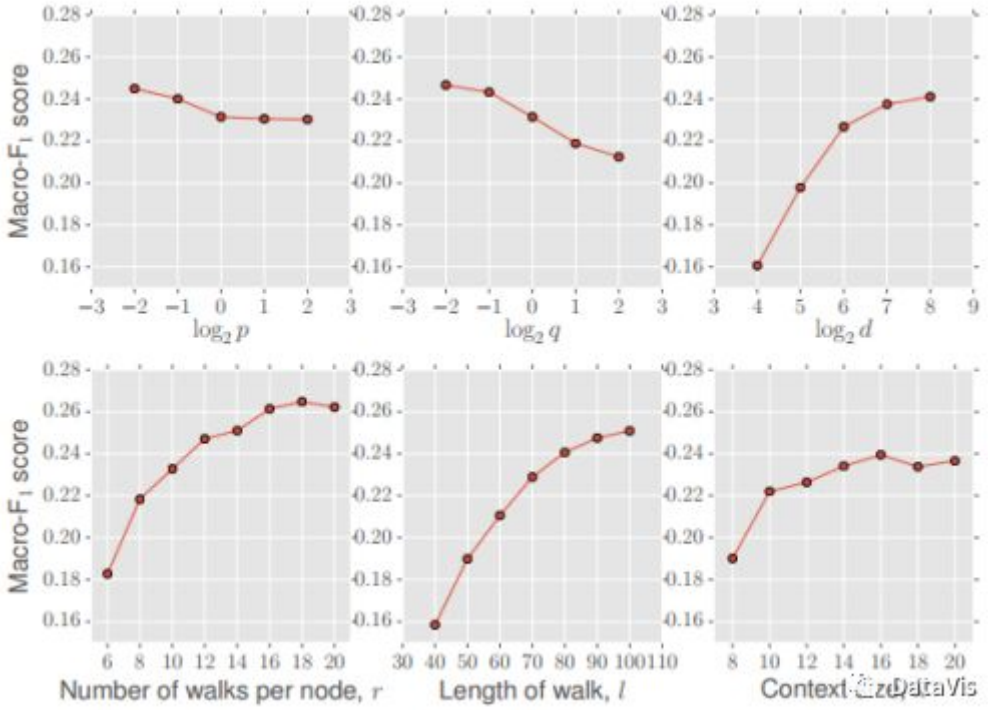
该实验在BlogCatalog、Protein-Protein Interactions （PPI）、Wikipedia三个网络中进行，对比了不同方法的效果。



▲x轴表示标记数据的分数，而顶行和底行中的y轴分别表示Micro-F1和Macro-F1分数。除DeepWalk和node2vec在PPI网络上具有可比性外，在其他网络中node2vec表现最佳。

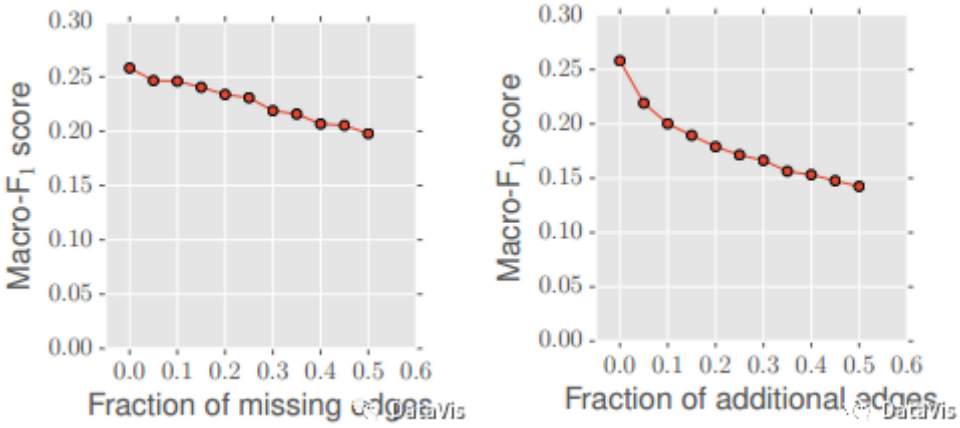
⑶ 参数灵敏度分析

由于node2vec中涉及参数较多，因此作者们对控制模型中各变量，对主要参数进行性能测试。



⑷ 干扰分析

作者们在BlogCatalog网络中，对链接缺失和具有噪声链接这两种不同情况进行node2vec的性能测试。



(5) 链路预测

该实验在Facebook、PPI、arXiv ASTRO-PH三个网络中进行，呈现了不同方法的在Average、Hadamard、Weighted-Li等评估标准下的表现。

Op	Algorithm	Dataset		
		Facebook	PPI	arXiv
	Common Neighbors	0.8100	0.7142	0.8153
	Jaccard's Coefficient	0.8880	0.7018	0.8067
	Adamic-Adar	0.8289	0.7126	0.8315
	Pref. Attachment	0.7137	0.6670	0.6996
(a)	Spectral Clustering	0.5960	0.6588	0.5812
	DeepWalk	0.7238	0.6923	0.7066
	LINE	0.7029	0.6330	0.6516
	node2vec	0.7266	0.7543	0.7221
(b)	Spectral Clustering	0.6192	0.4920	0.5740
	DeepWalk	**0.9680**	0.7441	0.9340
	LINE	0.9490	0.7249	0.8902
	node2vec	**0.9680**	**0.7719**	**0.9366**
(c)	Spectral Clustering	0.7200	0.6356	0.7099
	DeepWalk	0.9574	0.6026	0.8282
	LINE	0.9483	0.7024	0.8809
	node2vec	0.9602	0.6292	0.8468
(d)	Spectral Clustering	0.7107	0.6026	0.6765
	DeepWalk	0.9584	0.6118	0.8305
	LINE	0.9460	0.7106	0.8862
	node2vec	0.9606	0.6236	0.8497

▲总体而言，node2vec的Hadamard运算非常稳定，并几乎在所有网络上提供最佳性能。

因此，该文主要贡献有：

1. 基于word2vec提出node2vec，可以有效保留邻居节点的信息，高效提取网络中节点特征，具有较好的性能表现；

2. 有效结合BFS和DFS提出2nd-order random walks策略，通过参数调整使node2vec可灵活适应于不同的网络。

END