

Beijing Forest Studio
北京理工大学信息系统及安全对抗实验中心



网络表示学习

李东超 硕士研究生

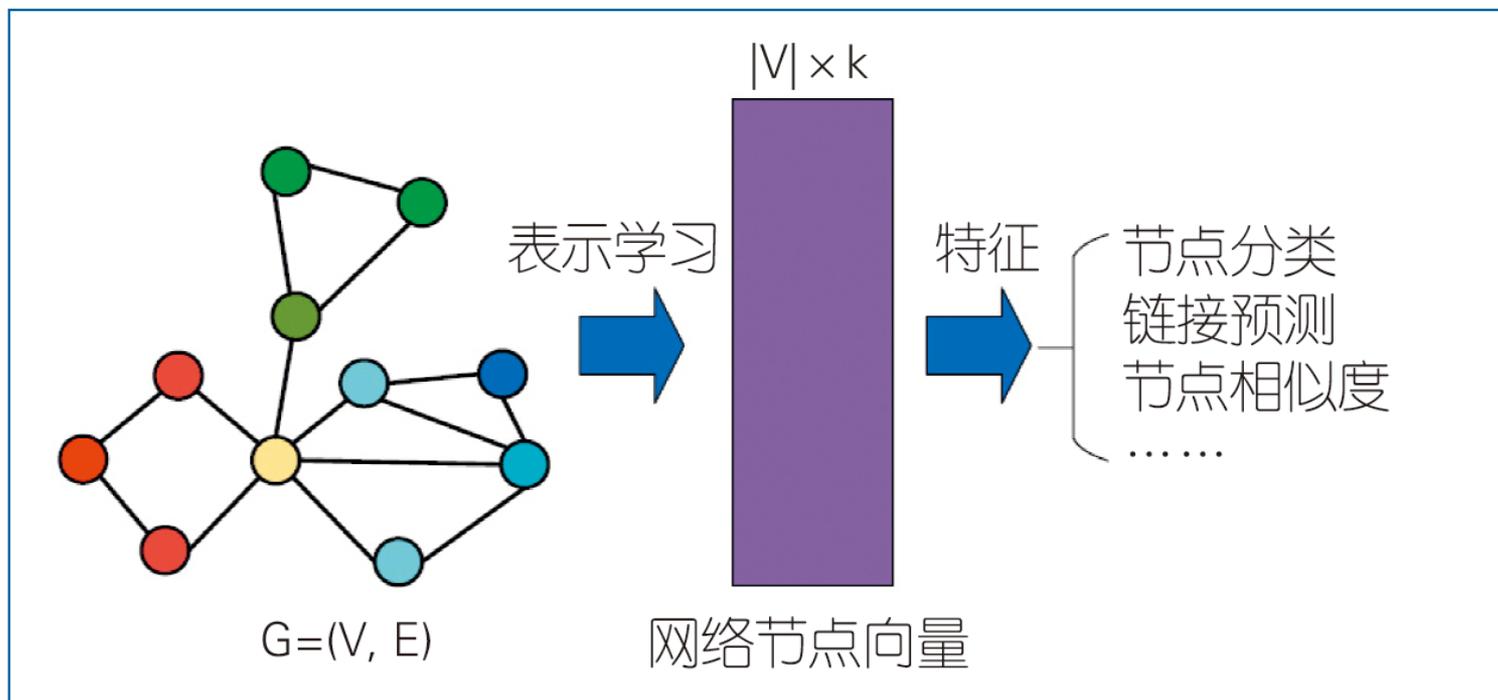
2018年09月16日

- 背景简介
- 基本概念
- 算法原理
- 优劣分析
- 应用总结
- 参考文献

- 预期收获
 - 熟悉网络表示学习的基本概念和分类
 - 理解LINE算法的基本原理
 - 了解网络表示学习的应用

• 网络表示学习用来做什么？ Network Embedding

网络表示学习算法负责从网络数据中学习得到网络中每个节点的向量表示, 之后这些节点表示就可以作为节点的特征应用于后续的网络应用任务, 如节点分类、链接预测等。



• 有哪些网络表示学习方法?



- KL散度

KL散度也叫交叉熵、相对熵，是一种量化两种概率分布 $P(x)$ 和 $Q(x)$ 之间差异的方式，用 $D(P||Q)$ 表示KL距离。当两个概率分布完全相同时，即 $P(x)=Q(x)$ ，其相对熵为0。

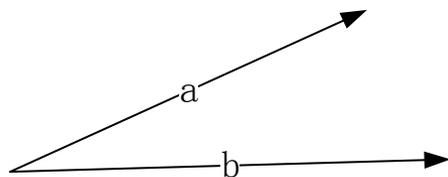
$$D(P||Q) = \sum_{x \in X} P(x) \log \frac{P(x)}{Q(x)}$$

• 向量内积

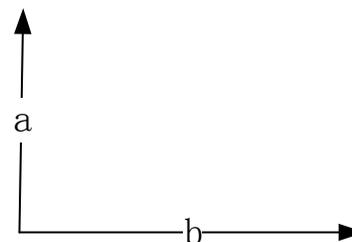
➤ $\mathbf{a} = [a_1, a_2, \dots, a_n]$ $\mathbf{b} = [b_1, b_2, \dots, b_n]$

➤ $\mathbf{a} \cdot \mathbf{b} = a_1 b_1 + a_2 b_2 + \dots + a_n b_n$

➤ $\mathbf{a} \cdot \mathbf{b} = |\mathbf{a}| |\mathbf{b}| \cos \theta$



a与b相似



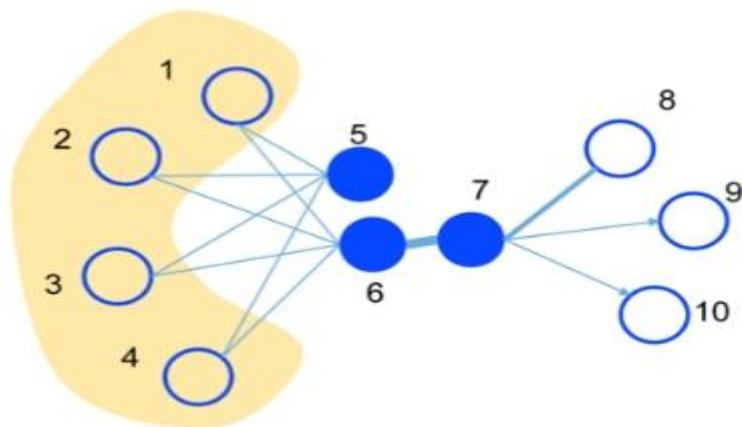
a与b不相似

所以向量内积可以用来表示向量之间的相似度

T	基于网络连接信息，完成每个节点的向量表示
I	网络节点、边
P	1. 初始化节点表示向量 2. 根据目标函数迭代更新向量表示
O	每个节点的低维稠密表示向量

P	如何确定生成的节点表示向量更贴近真实情况
C	节点之间存在联系，构成网络
D	目标函数的构建
L	CCF A类会议

- LINE算法
- 节点6和节点7直接相连，而且连接边的权重值很大（线很粗），所以节点6和节点7之间存在很强的一阶相似性，它们在嵌入空间中的表示应该很接近。节点5和节点6之间虽然没有直接连接，但是由于它们有很多共同的邻居节点（阴影），所以节点5和节点6之间存在二阶相似性，它们在嵌入空间中的表示也应该很接近。



- 一阶相似性

一阶相似性定义为：每条无向边 (i, j) 的权重 w_{ij} ，如果节点之间没有连接，一阶相似性为0。

对于每条无向边 (i, j) ，节点 i 和节点 j 之间的联合概率分布定义为：

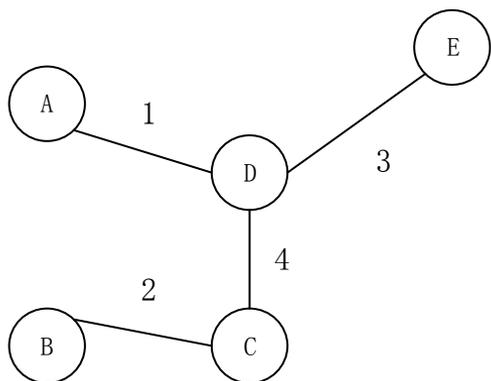
$$p_1(i, j) = \frac{1}{1 + \exp(-\mathbf{u}_i^T \cdot \mathbf{u}_j)}$$

其中， \mathbf{u}_i 是节点的低维度特征向量表示

- 经验概率分布定义为该条边在所有边中的比重大小

$$p'_1(i, j) = \frac{w_{ij}}{W}$$

其中 $W = \sum_{(i,j) \in E} w_{ij}$



$$w_{AD} = 1 \quad w_{BC} = 2$$
$$w_{DE} = 3 \quad w_{CD} = 4$$

$$W = w_{AD} + w_{BC} + w_{DE} + w_{CD}$$

- 构建一阶相似性的目标函数
- 通过最小化目标函数 $O_1 = d(p'_1, p_1)$ ，其中 $d(\dots)$ 表示两种分布间的距离，通过KL散度来计算它们之间的距离

$$D(P||Q) = \sum_{x \in X} P(x) \log \frac{P(x)}{Q(x)}$$

$$O_1 = \sum_{(i,j) \in E} \frac{w_{ij}}{W} \log \frac{w_{ij}/W}{p_1}$$

$$O_1 = \sum_{(i,j) \in E} \frac{w_{ij}}{W} \log \frac{w_{ij}/W}{p_1}$$

$$O_1 = \sum_{(i,j) \in E} w_{ij} \log \frac{w_{ij}}{p_1}$$

$$O_1 = \sum_{(i,j) \in E} w_{ij} \log w_{ij} - w_{ij} \log p_1$$

$$O_1 = - \sum_{(i,j) \in E} w_{ij} \log p_1$$

- 二阶相似性
- 二阶相似性是指一对顶点对之间的相似性和它们邻居网络结构之间的相似性是相似的。从数学的角度定义， $q_i = (w_{i1}, w_{i2}, \dots, w_{i|V|})$ 表示节点*i*与其他所有节点的一阶相似性，节点*i*与节点*j*之间的二阶相似性定义为 q_i 和 q_j 之间的相似性。

- 对每个节点维护两个向量， u_i' 和 u_i 。其中 u_i' 表示节点 i 作为邻居节点时的向量， u_i 表示节点 i 作为节点本身的向量。
- 定义节点 j 是节点 i 邻居的概率为

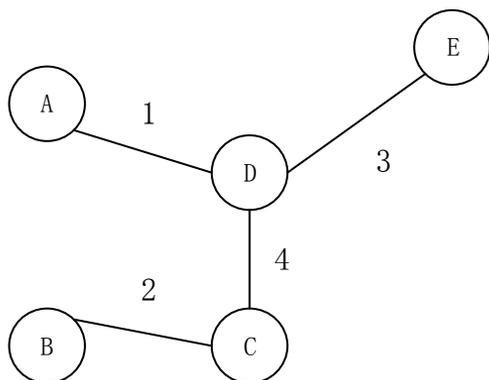
$$p_2(j|i) = \frac{\exp(u_j^{T'} \cdot u_i)}{\sum_{k=1}^{|V|} \exp(u_k^{T'} \cdot u_i)}$$

- j 和 i 越相似，对应向量点积越大， j 是 i 的邻居的概率越大

- 经验概率

$$p'_2(j|i) = \frac{w_{ij}}{d_i}$$

其中, $d_i = \sum_{k \in N(i)} w_{ik}$, 表示顶点*i*的出度

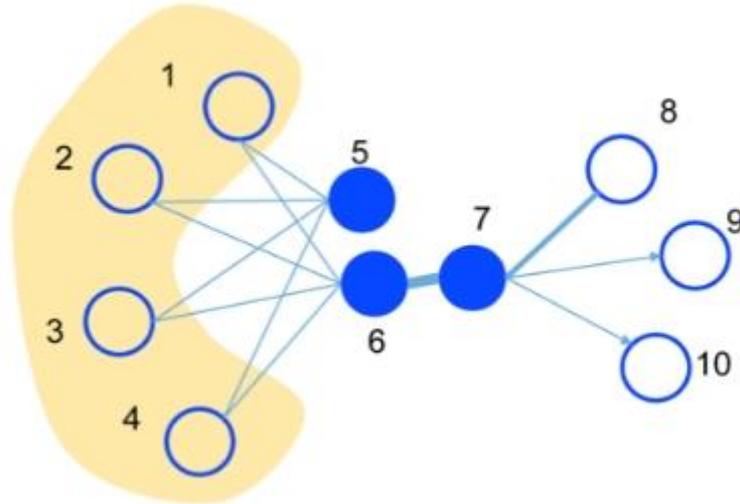


$$d_D = 1 + 3 + 4$$

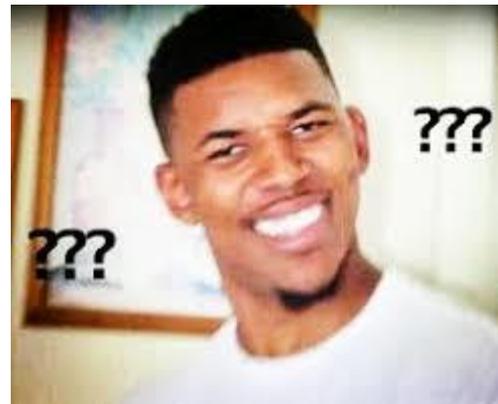
- 目标函数
- 通过最小化目标函数 $O_1 = d(p'_1, p_1)$, 其中 $d(.,.)$ 表示两种分布间的距离, 通过KL散度来计算它们之间的距离

$$O_2 = - \sum_{(i,j) \in E} w_{ij} \log p_2(j|i)$$

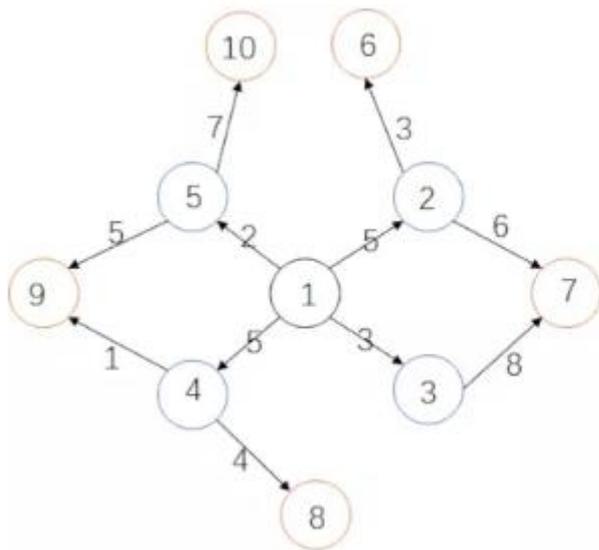
二阶相似性



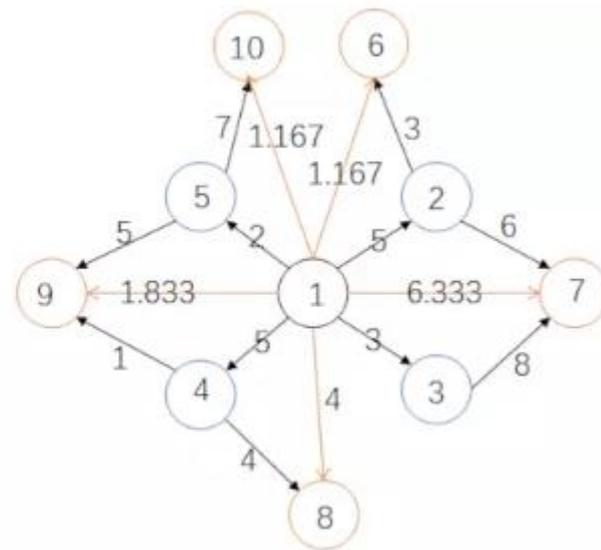
$$w_{56} = 0$$



reconstruct.cpp



(a) 原图



(b) 重构图

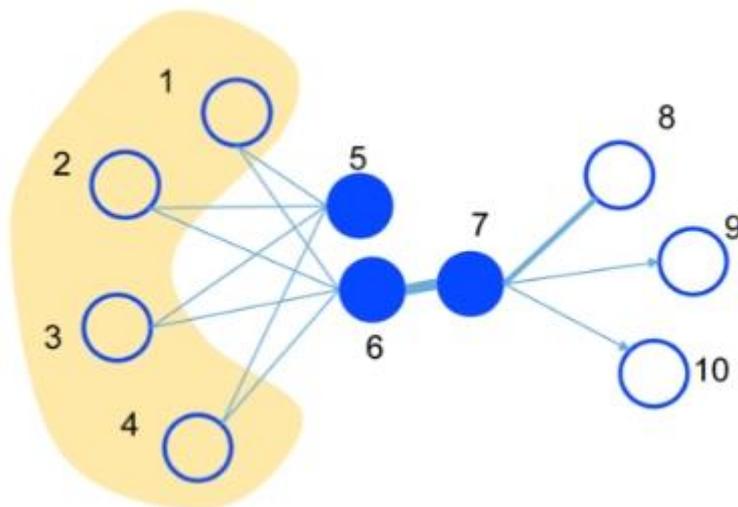
$$w(a, n) = w(a, m) * \left(\frac{w(m, n)}{w(m)} \right)$$

其中， $w(m)$ 是所有起点为 m 的边的权值之和

比如：节点1、6之间的边的权值为 $5 * (3/9) = 1.67$ ；节点1、7之间的边的权值为 $5 * (6/9) + 3 * (8/8) = 6.333$

$$p_2(j|i) = \frac{\exp(u_j^{T'} \cdot u_i)}{\sum_{k=1}^{|V|} \exp(u_k^{T'} \cdot u_i)}$$

- O_2 的计算代价十分昂贵，所以优化时使用了负采样方法



为每条边定义新的 $p_2(j|i)$

$$\log \sigma(u_j'^T \cdot u_i) + \sum_{i=1}^K E_{v \sim p_n(v)} [\log \sigma(-u_n'^T \cdot u_i)]$$

- 边采样

$$O_2 = - \sum_{(i,j) \in E} w_{ij} \log p_2(j|i)$$

$$\frac{\partial O_2}{\partial i} = w_{ij} \frac{\partial \log p_2(j|i)}{\partial i}$$

- 当边缘的权重差异很大时，会出现问题
- 解决方法：从原始边缘进行采样并将采样的边缘视为二进制边缘，其中采样概率与原始边缘权重成比例。通过这种边缘采样处理，整体目标函数保持不变。该问题归结为如何根据其权重对边缘进行采样。

- 语言网络
- 在词类比任务中从整套英文维基百科页面构建一个词语共现网络。每5个字滑动窗口内的字被认为是共现的，出现频率小于5的词被滤除。在文档分类任务中，使用维基百科界面的摘要，及其对应的类别。

Metric	Algorithm	10%	20%	30%	40%	50%	60%	70%	80%	90%
Micro-F1	GF	79.63	80.51	80.94	81.18	81.38	81.54	81.63	81.71	81.78
	DeepWalk	78.89	79.92	80.41	80.69	80.92	81.08	81.21	81.35	81.42
	SkipGram	79.84	80.82	81.28	81.57	81.71	81.87	81.98	82.05	82.09
	LINE-SGD(1st)	76.03	77.05	77.57	77.85	78.08	78.25	78.39	78.44	78.49
	LINE-SGD(2nd)	74.68	76.53	77.54	78.18	78.63	78.96	79.19	79.40	79.57
	LINE(1st)	79.67	80.55	80.94	81.24	81.40	81.52	81.61	81.69	81.67
	LINE(2nd)	79.93	80.90	81.31	81.63	81.80	81.91	82.00	82.11	82.17
	LINE(1st+2nd)	81.04**	82.08**	82.58**	82.93**	83.16**	83.37**	83.52**	83.63**	83.74**
Macro-F1	GF	79.49	80.39	80.82	81.08	81.26	81.40	81.52	81.61	81.68
	DeepWalk	78.78	79.78	80.30	80.56	80.82	80.97	81.11	81.24	81.32
	SkipGram	79.74	80.71	81.15	81.46	81.63	81.78	81.88	81.98	82.01
	LINE-SGD(1st)	75.85	76.90	77.40	77.71	77.94	78.12	78.24	78.29	78.36
	LINE-SGD(2nd)	74.70	76.45	77.43	78.09	78.53	78.83	79.08	79.29	79.46
	LINE(1st)	79.54	80.44	80.82	81.13	81.29	81.43	81.51	81.60	81.59
	LINE(2nd)	79.82	80.81	81.22	81.52	81.71	81.82	81.92	82.00	82.07
	LINE(1st+2nd)	80.94**	81.99**	82.49**	82.83**	83.07**	83.29**	83.42**	83.55**	83.66**

Significantly outperforms GF at the: ** 0.01 and * 0.05 level, paired t-test.

- 社交网络
- 在社交网络中将每个用户节点分配到一个或多个社区，通过多标签分类任务来评估算法。实验中，随机抽样不同比例的顶点进行训练，其余部分用于评估。

Metric	Algorithm	1%	2%	3%	4%	5%	6%	7%	8%	9%	10%
Micro-F1	GF	25.43 (24.97)	26.16 (26.48)	26.60 (27.25)	26.91 (27.87)	27.32 (28.31)	27.61 (28.68)	27.88 (29.01)	28.13 (29.21)	28.30 (29.36)	28.51 (29.63)
	DeepWalk	39.68	41.78	42.78	43.55	43.96	44.31	44.61	44.89	45.06	45.23
	DeepWalk(256dim)	39.94	42.17	43.19	44.05	44.47	44.84	45.17	45.43	45.65	45.81
	LINE(1st)	35.43 (36.47)	38.08 (38.87)	39.33 (40.01)	40.21 (40.85)	40.77 (41.33)	41.24 (41.73)	41.53 (42.05)	41.89 (42.34)	42.07 (42.57)	42.21 (42.73)
	LINE(2nd)	32.98 (36.78)	36.70 (40.37)	38.93 (42.10)	40.26 (43.25)	41.08 (43.90)	41.79 (44.44)	42.28 (44.83)	42.70 (45.18)	43.04 (45.50)	43.34 (45.67)
	LINE(1st+2nd)	39.01* (40.20)	41.89 (42.70)	43.14 (43.94**)	44.04 (44.71**)	44.62 (45.19**)	45.06 (45.55**)	45.34 (45.87**)	45.69** (46.15**)	45.91** (46.33**)	46.08** (46.43**)
Macro-F1	GF	7.38 (11.01)	8.44 (13.55)	9.35 (14.93)	9.80 (15.90)	10.38 (16.45)	10.79 (16.93)	11.21 (17.38)	11.55 (17.64)	11.81 (17.80)	12.08 (18.09)
	DeepWalk	28.39	30.96	32.28	33.43	33.92	34.32	34.83	35.27	35.54	35.86
	DeepWalk(256dim)	28.95	31.79	33.16	34.42	34.93	35.44	35.99	36.41	36.78	37.11
	LINE(1st)	28.74 (29.40)	31.24 (31.75)	32.26 (32.74)	33.05 (33.41)	33.30 (33.70)	33.60 (33.99)	33.86 (34.26)	34.18 (34.52)	34.33 (34.77)	34.44 (34.92)
	LINE(2nd)	17.06 (22.18)	21.73 (27.25)	25.28 (29.87)	27.36 (31.88)	28.50 (32.86)	29.59 (33.73)	30.43 (34.50)	31.14 (35.15)	31.81 (35.76)	32.32 (36.19)
	LINE(1st+2nd)	29.85 (29.24)	31.93 (33.16**)	33.96 (35.08**)	35.46** (36.45**)	36.25** (37.14**)	36.90** (37.69**)	37.48** (38.30**)	38.10** (38.80**)	38.46** (39.15**)	38.82** (39.40**)

Significantly outperforms DeepWalk at the: ** 0.01 and * 0.05 level, paired t-test.

- 引用网络
- 使用DBLP数据集来构建作者和文章两个引用网络。引用网络中边的权值表示一位作者引用另一位作者的论文数量。

Metric	Algorithm	10%	20%	30%	40%	50%	60%	70%	80%	90%
Micro-F1	DeepWalk	63.98	64.51	64.75	64.81	64.92	64.99	64.99	65.00	64.90
	LINE-SGD(2nd)	56.64	58.95	59.89	60.20	60.44	60.61	60.58	60.73	60.59
	LINE(2nd)	62.49	63.30	63.63	63.77	63.84	63.94	63.96	64.00	63.77
		(64.69*)	(65.47**)	(65.85**)	(66.04**)	(66.19**)	(66.25**)	(66.30**)	(66.12**)	(66.05**)
Macro-F1	DeepWalk	63.02	63.60	63.84	63.90	63.98	64.06	64.09	64.11	64.05
	LINE-SGD(2nd)	55.24	57.63	58.56	58.82	59.11	59.27	59.28	59.46	59.37
	LINE(2nd)	61.43	62.38	62.73	62.87	62.93	63.05	63.07	63.13	62.95
		(63.49*)	(64.42**)	(64.84**)	(65.05**)	(65.19**)	(65.26**)	(65.29**)	(65.14**)	(65.14**)

Significantly outperforms DeepWalk at the: ** 0.01 and * 0.05 level, paired t-test.

- **优势**
 - 目标函数设计巧妙，保留了一阶和二阶的接近度，相互互补
 - 解决了加权边缘随机梯度下降的限制，而不影响效率
- **劣势**
 - 当网络中添加新顶点时，如果为观察到新顶点与现有顶点的连接，则无法给出新顶点的向量表示

- 节点分类
- 链接预测
- 社区发现
- 推荐系统
-

- [1] Tang J, Qu M, Wang M, et al. LINE: Large-scale Information Network Embedding[J]. 2015, 2(2):1067–1077.
- [2] Mikolov T, Sutskever I, Chen K, et al. Distributed representations of words and phrases and their compositionality [C] // Advances in neural information processing systems. 2013: 3111–3119.
- [3] 涂存超, 杨成, 刘知远, 等. 网络表示学习综述[J]. 中国科学: 信息科学, 2017(8).

知人者智，自知者明。
胜人者有力，自胜者
强。知足者富。强行
者有志。不失其所者
久。死而不亡者，寿。

谢谢！

