# node2vec: Scalable Feature Learning for Networks

刘希阳

8-3

## 1 INTRODUCTION

- 利用网络有很多应用场景
  - 根据蛋白质分子预测它的功能
  - 发现基因之间可能存在的交互
  - 预测可能存在的好友

- 以上的任务依赖于学习图中节点的特征。但是之前的方法会依赖人工的特征工程
- 现在存在一些自动学习图特征的方法，但是如果是针对特定任务的方法通用性不够；对于无监督方法来说，现存的特征学习方法还不能足够的捕捉出显示网络中被观测到的联通模式的的多样性

## 1 INTRODUCTION

- 因此，本文就提出了一种无监督的方法
- 核心思想：通过特定的游走方式进行采样，对于每个点都会生成对应的序列。再将这些序列视为文本导入skip-gram模型，即可得到每个节点的向量

## 2 FEATURE LEARNING FRAMEWORK

- Objective function

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

f(u)是节点u的特征（embedding）；$N_S(u)$是节点u的邻居节点

- Conditional independence

$$Pr(N_S(u)|f(u)) = \prod_{n_i \in N_S(u)} Pr(n_i|f(u))$$ 假设邻居节点之间互相独立

- Symmetry in feature space

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

## 2 FEATURE LEARNING FRAMEWORK

- Final equation

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

**How to find $\boldsymbol{N_S(u)}$ ?**

## 2 FEATURE LEARNING FRAMEWORK

- BFS&DFS

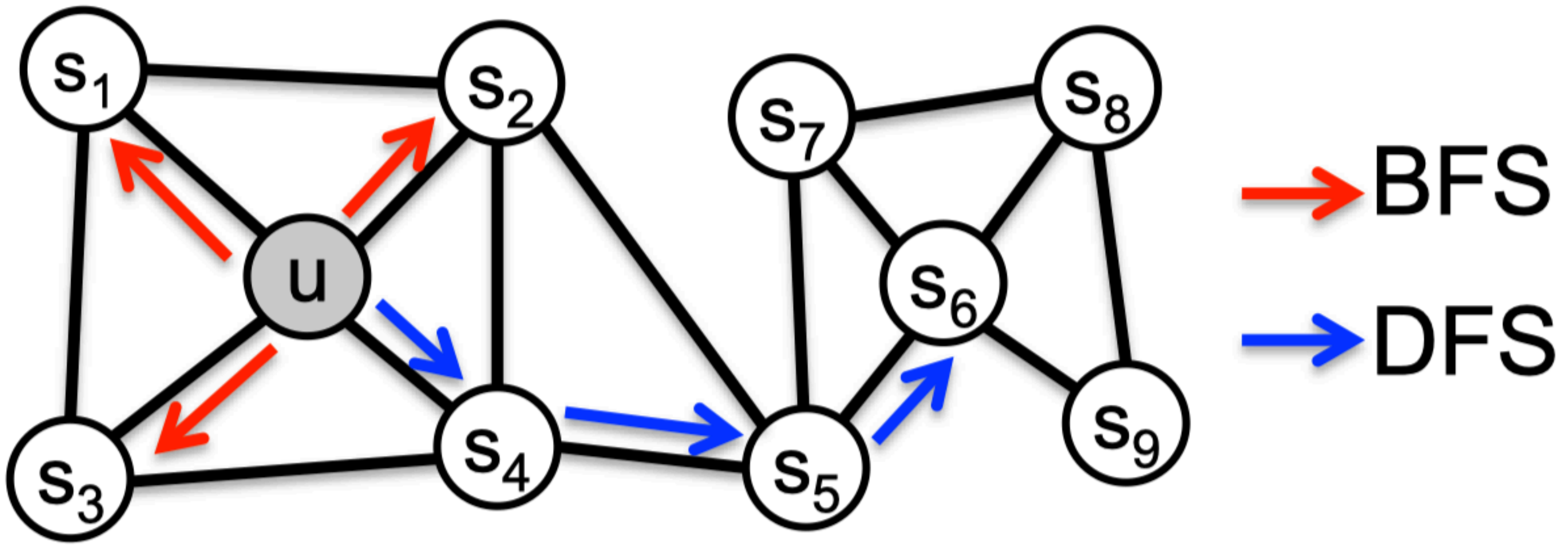
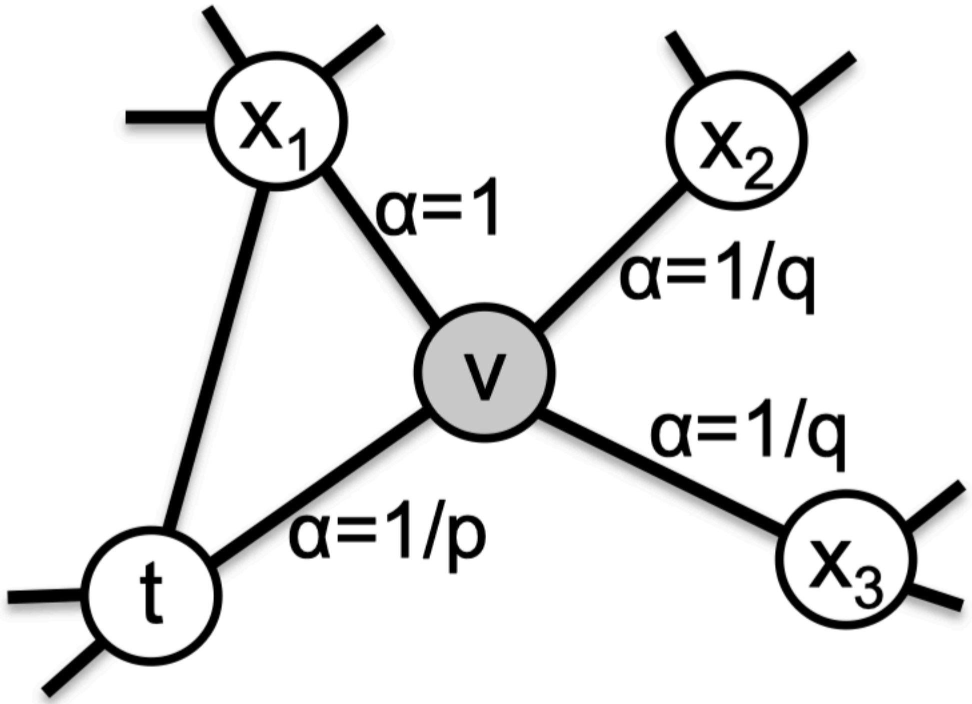


Figure 1: BFS and DFS search strategies from node $u$ ($k = 3$).

## 2 FEATURE LEARNING FRAMEWORK

- 2-order Biased Random walk



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

Now, just traversed edge (t, v) and now resides at node v

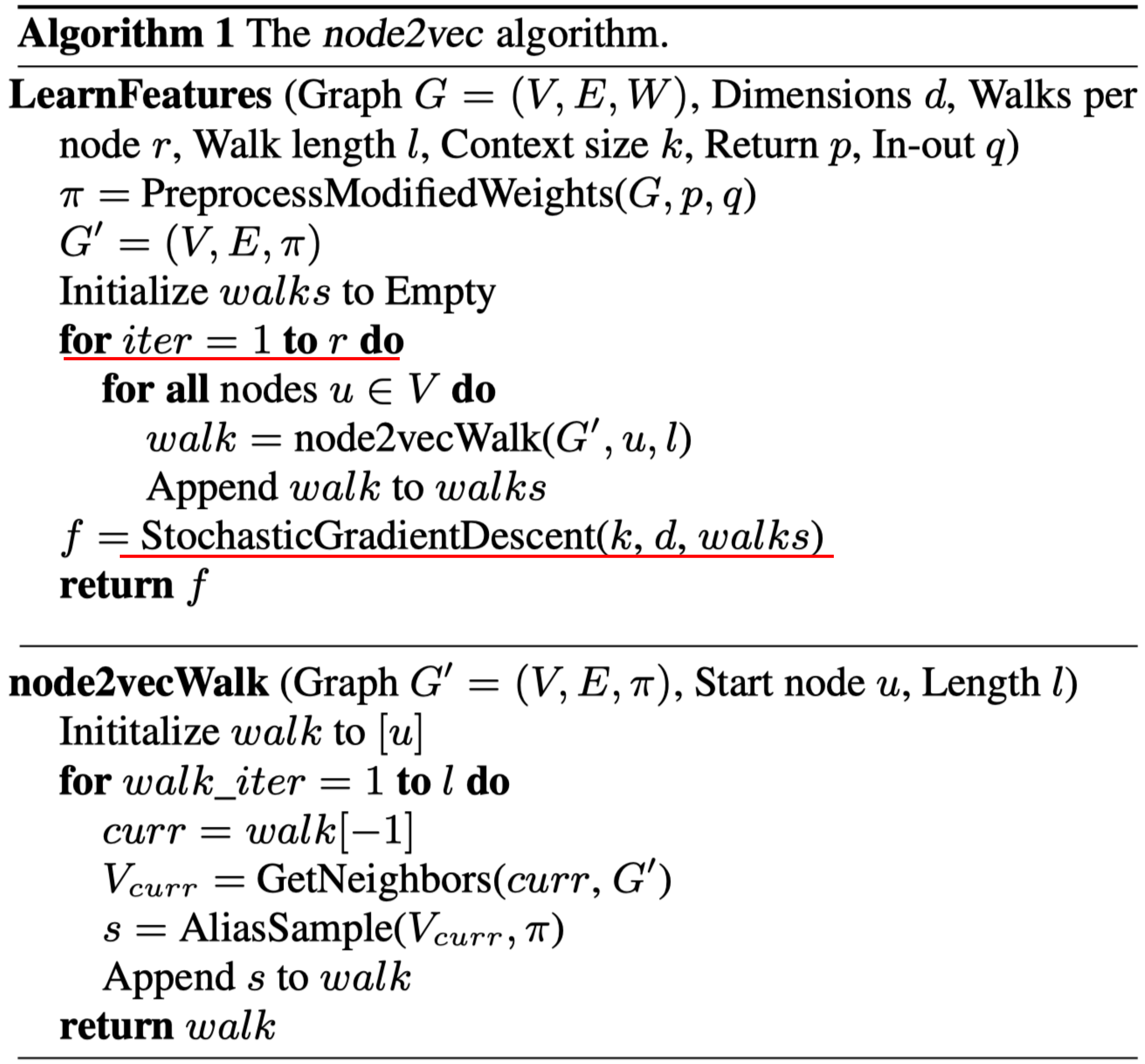
$$\pi_{vx} = \alpha_{pq}(t, x) \cdot w_{vx}$$

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

$d_{t,x} \in \{0,1,2\}$

# 2 FEATURE LEARNING FRAMEWORK

- Node2vec:



**Algorithm 1** The *node2vec* algorithm.

```
LearnFeatures (Graph $G = (V, E, W)$, Dimensions $d$, Walks per
    node $r$, Walk length $l$, Context size $k$, Return $p$, In-out $q$)
    $\pi$ = PreprocessModifiedWeights($G, p, q$)
    $G' = (V, E, \pi)$
    Initialize $walks$ to Empty
    for $iter = 1$ to $r$ do
        for all nodes $u \in V$ do
            $walk$ = node2vecWalk($G', u, l$)
            Append $walk$ to $walks$
    $f$ = StochasticGradientDescent($k$, $d$, $walks$)
    return $f$
```

```
node2vecWalk (Graph $G' = (V, E, \pi)$, Start node $u$, Length $l$)
    Inititalize $walk$ to $[u]$
    for $walk\_iter = 1$ to $l$ do
        $curr = walk[-1]$
        $V_{curr}$ = GetNeighbors($curr$, $G'$)
        $s$ = AliasSample($V_{curr}, \pi$)
        Append $s$ to $walk$
    return $walk$
```

核心两步：

1. 进行r次有偏随机游走获得每个节点的context
2. 利用SGD更新节点的表示

## 3 EXPERIMENTS

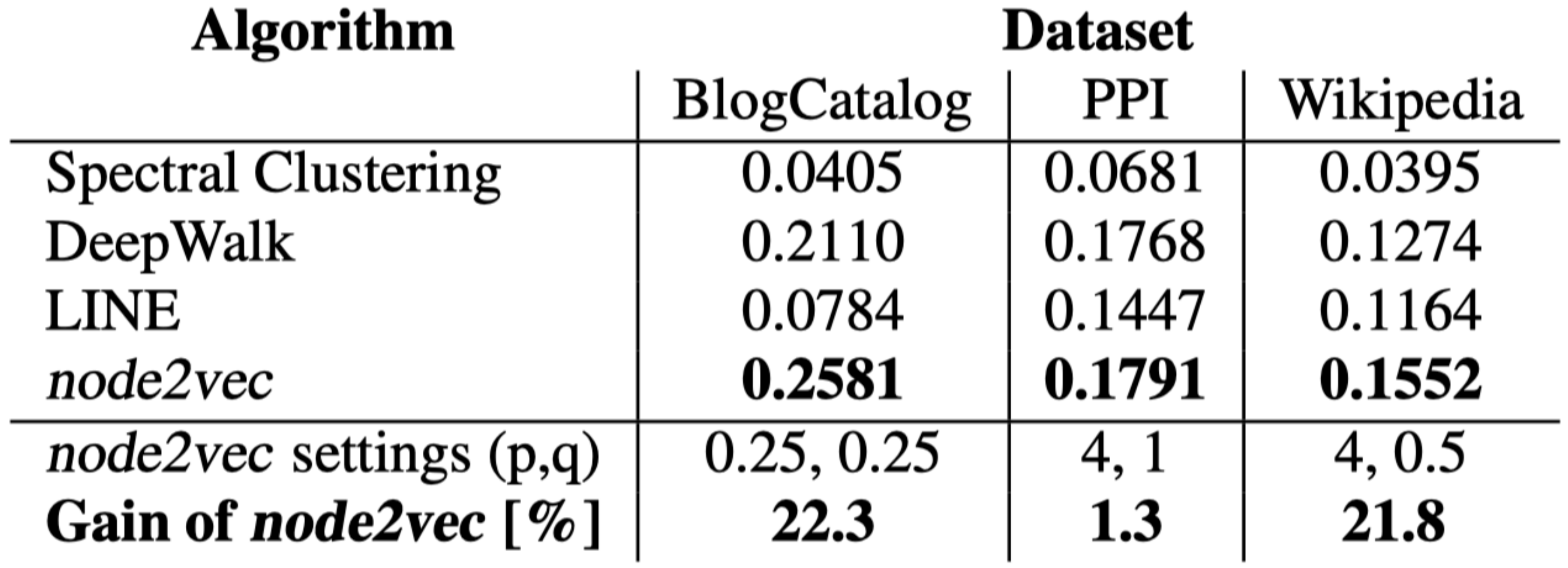
- 1. Multi-label classification

Dataset:

- BlogCatalog: social relationships of the bloggers listed on the BlogCatalog website.
- Protein-Protein Interactions (PPI): hallmark gene sets and represent biological states.
- Wikipedia: words appearing in the Wikipedia dump.

## 3 EXPERIMENTS

- 1. Multi-label classification
  - logistic regression classifier with L2 regularization.
  - 50-50 train/test

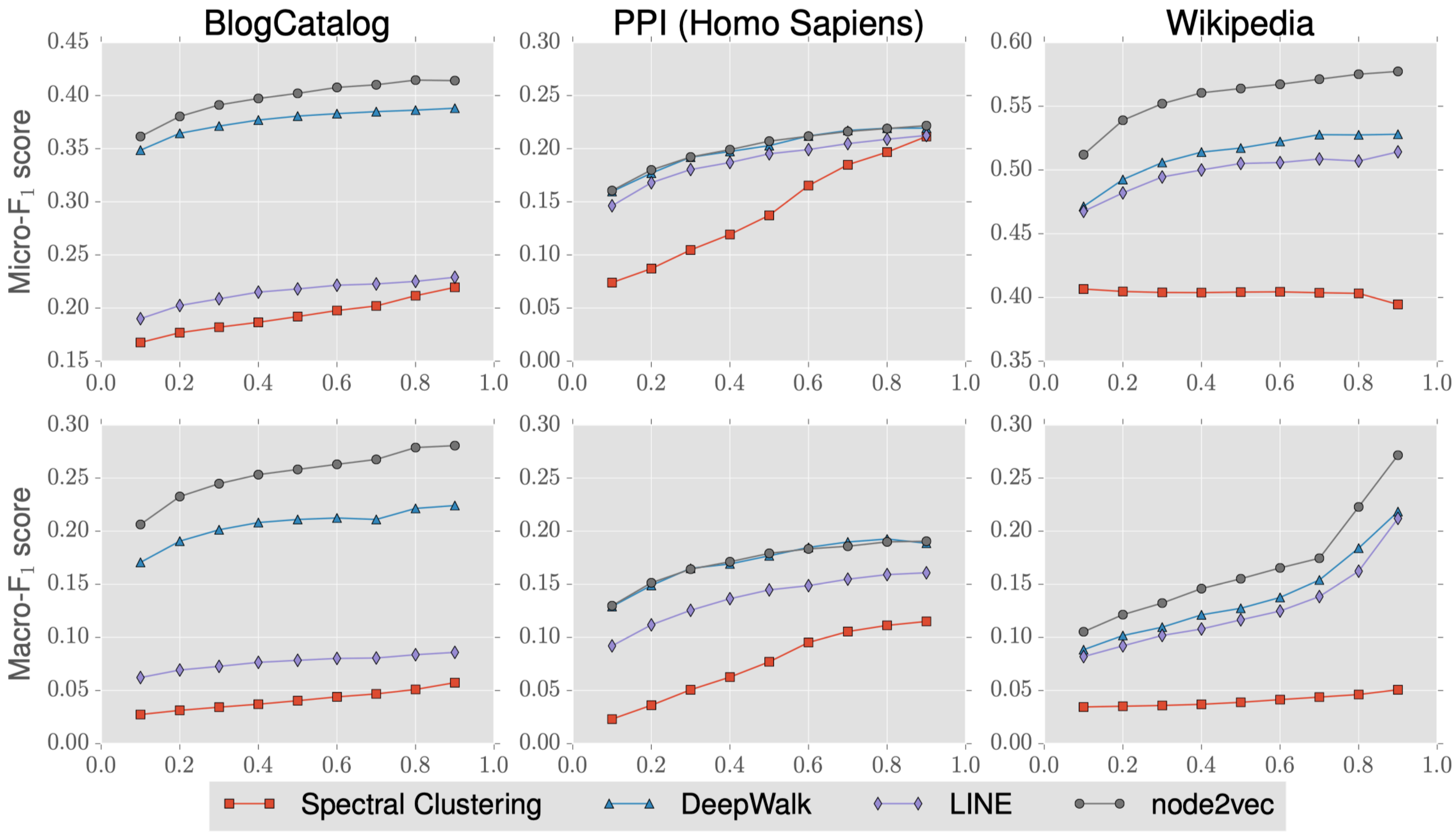


| **Algorithm** | **Dataset** | | |
|---|---|---|---|
| | BlogCatalog | PPI | Wikipedia |
| Spectral Clustering | 0.0405 | 0.0681 | 0.0395 |
| DeepWalk | 0.2110 | 0.1768 | 0.1274 |
| LINE | 0.0784 | 0.1447 | 0.1164 |
| *node2vec* | **0.2581** | **0.1791** | **0.1552** |
| *node2vec* settings (p,q) | 0.25, 0.25 | 4, 1 | 4, 0.5 |
| **Gain of *node2vec* [%]** | **22.3** | **1.3** | **21.8** |

Table 2: Macro-$F_1$ scores for multilabel classification on BlogCat-

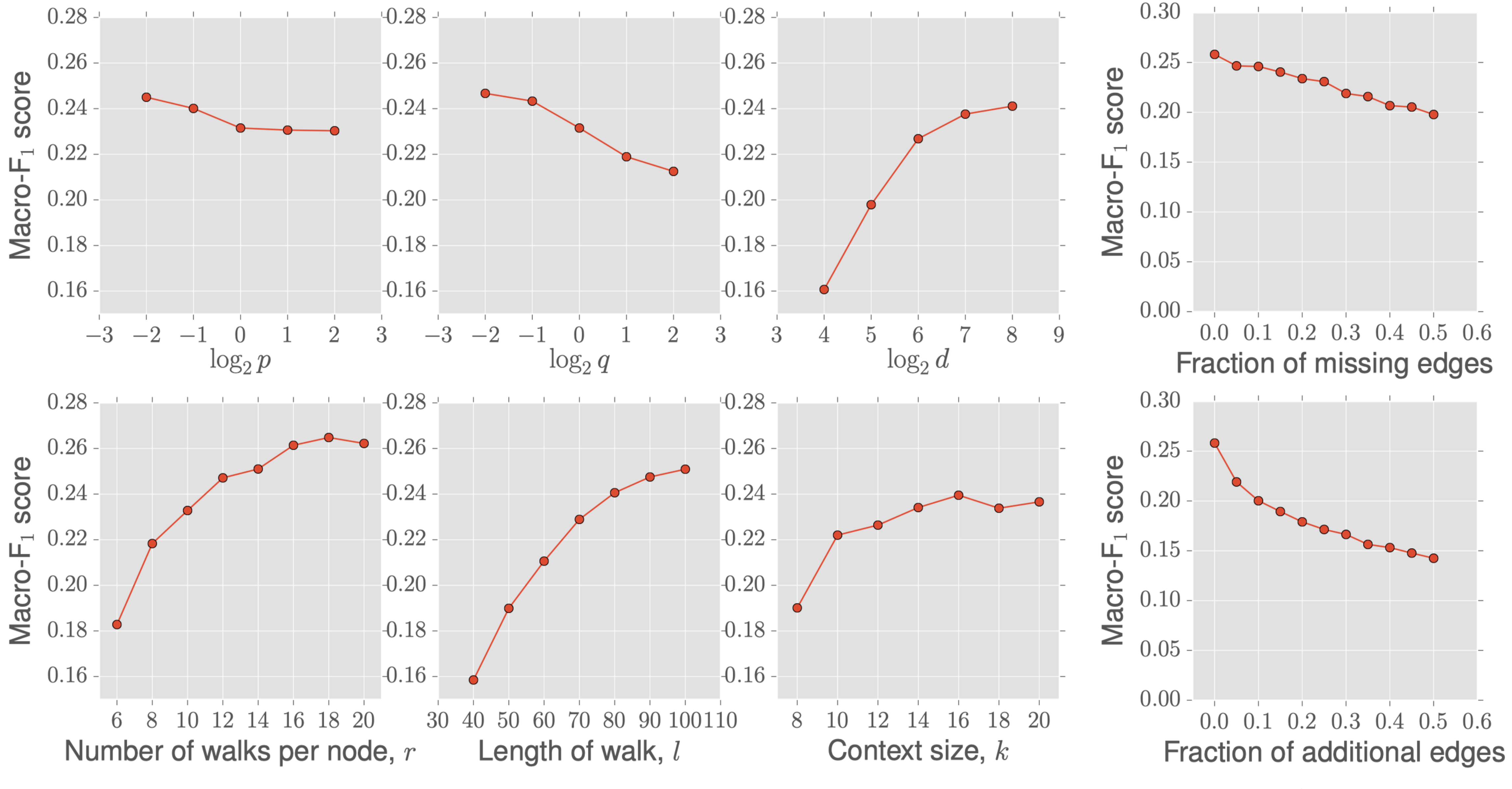
# 3 EXPERIMENTS

Varying the amount of labeled data used for training



train-test split from 10% to 90%

- 2&3. Parameter sensitivity and Perturbation Analysis
  - BlogCatalog, 50-50 split between labeled and unlabeled data.



(a)

(b)

# 3 EXPERIMENTS

- 4. Scalability
  - Erdos-Renyi graphs
  - 100 to 1,000,000 nodes and constant average degree of 10.

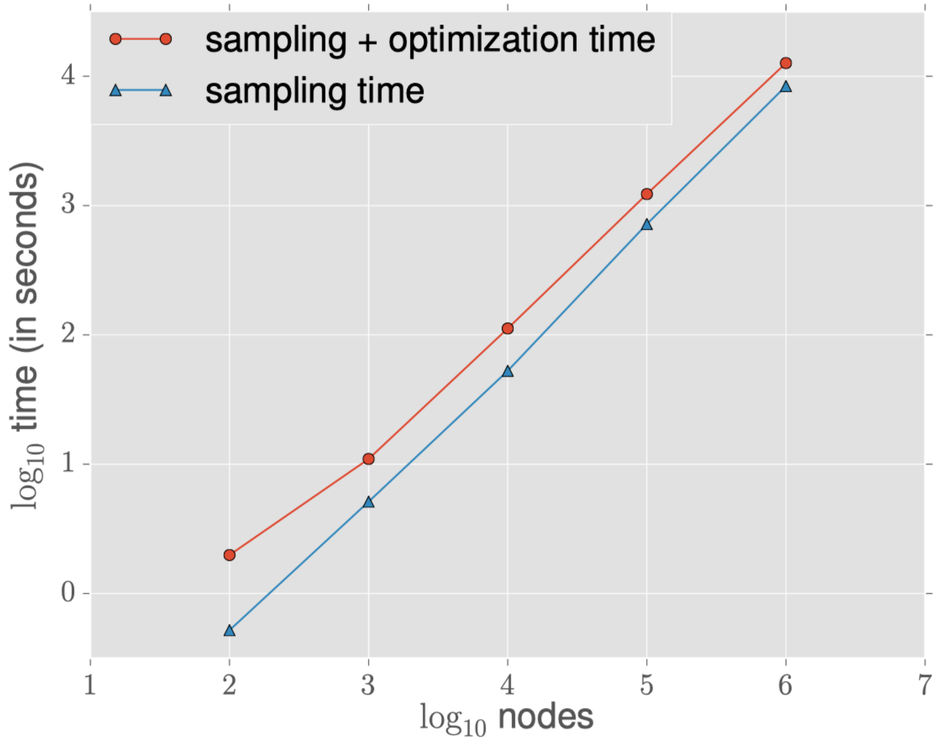
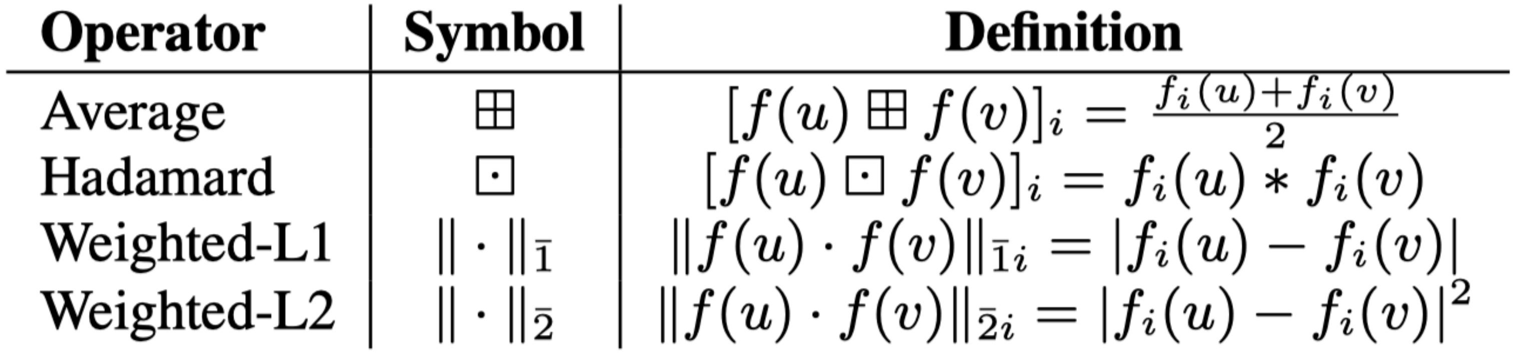


Figure 6: Scalability of *node2vec* on Erdos-Renyi graphs with an average degree of 10.

## 3 EXPERIMENTS

- 5. Link prediction
  - Edge features $(u, v)$
  - Given pair $f(u), f(v)$, generate $g(u, v)$ , a representation of edge.

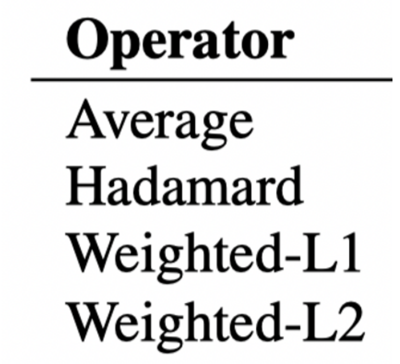


| Operator | Symbol | Definition |
|---|---|---|
| Average | $\boxplus$ | $[f(u) \boxplus f(v)]_i = \frac{f_i(u)+f_i(v)}{2}$ |
| Hadamard | $\boxdot$ | $[f(u) \boxdot f(v)]_i = f_i(u) * f_i(v)$ |
| Weighted-L1 | $\|\cdot\|_{\bar{1}}$ | $\|f(u) \cdot f(v)\|_{\bar{1}i} = \lvert f_i(u) - f_i(v)\rvert$ |
| Weighted-L2 | $\|\cdot\|_{\bar{2}}$ | $\|f(u) \cdot f(v)\|_{\bar{2}i} = \lvert f_i(u) - f_i(v)\rvert^2$ |

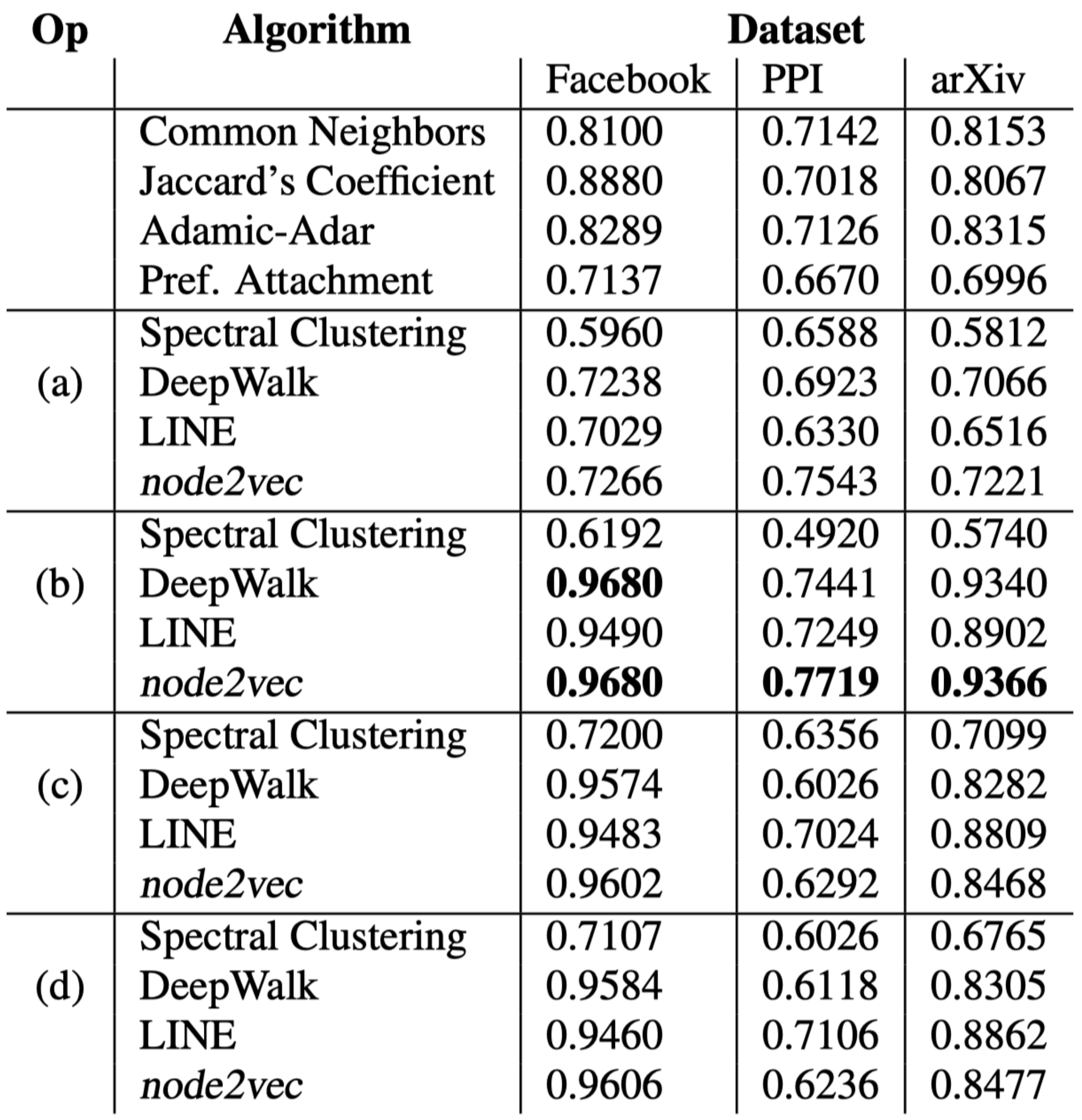
Table 1: Choice of binary operators $\circ$ for learning edge features. The definitions correspond to the $i$th component of $g(u, v)$.

## 3 EXPERIMENTS

- 5. Link prediction
  - Datasets:
    - Facebook
    - Protein-Protein Interactions (PPI)
    - ASTRO-PH



| Operator |
|---|
| Average |
| Hadamard |
| Weighted-L1 |
| Weighted-L2 |



| 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 | 0.8477 |

Table 4: Area Under Curve (AUC) scores for link prediction. Com-

## 7 CONCLUSION

- ❑ BFS and DFS.
- ❑ No independent embedding for edges.
- ❑ No deep architecture.

# THANKS

8-3