node2vec: Scalable Feature Learning for Networks

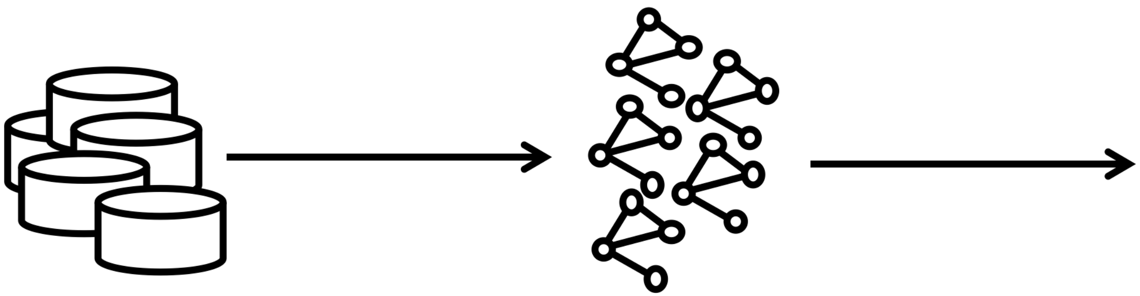
Aditya Grover, Jure Leskovec

Farzaneh Heidari

Outline

- word2vec (Background)
- Random Walk (Background)
- node2vec
- Evaluation Results
- Deficiencies

End-to-End Graph Analytics



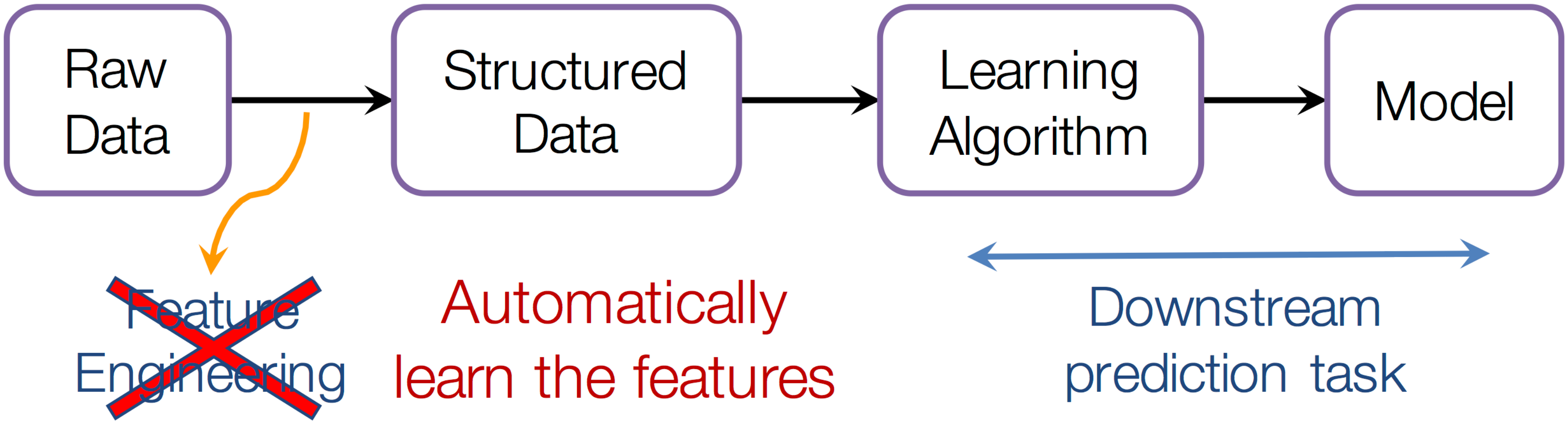
New knowledge and insights

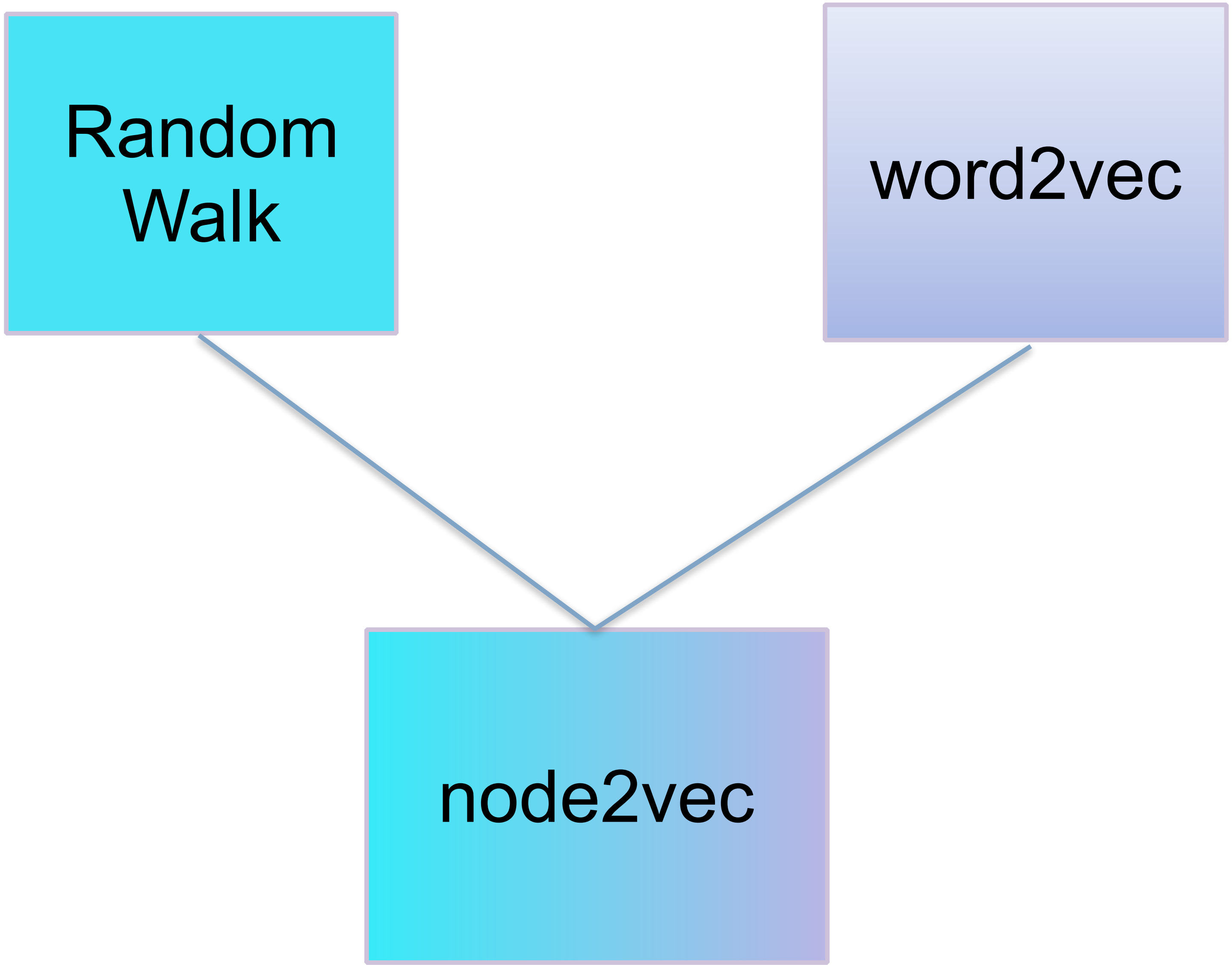
Data

Graph analytics

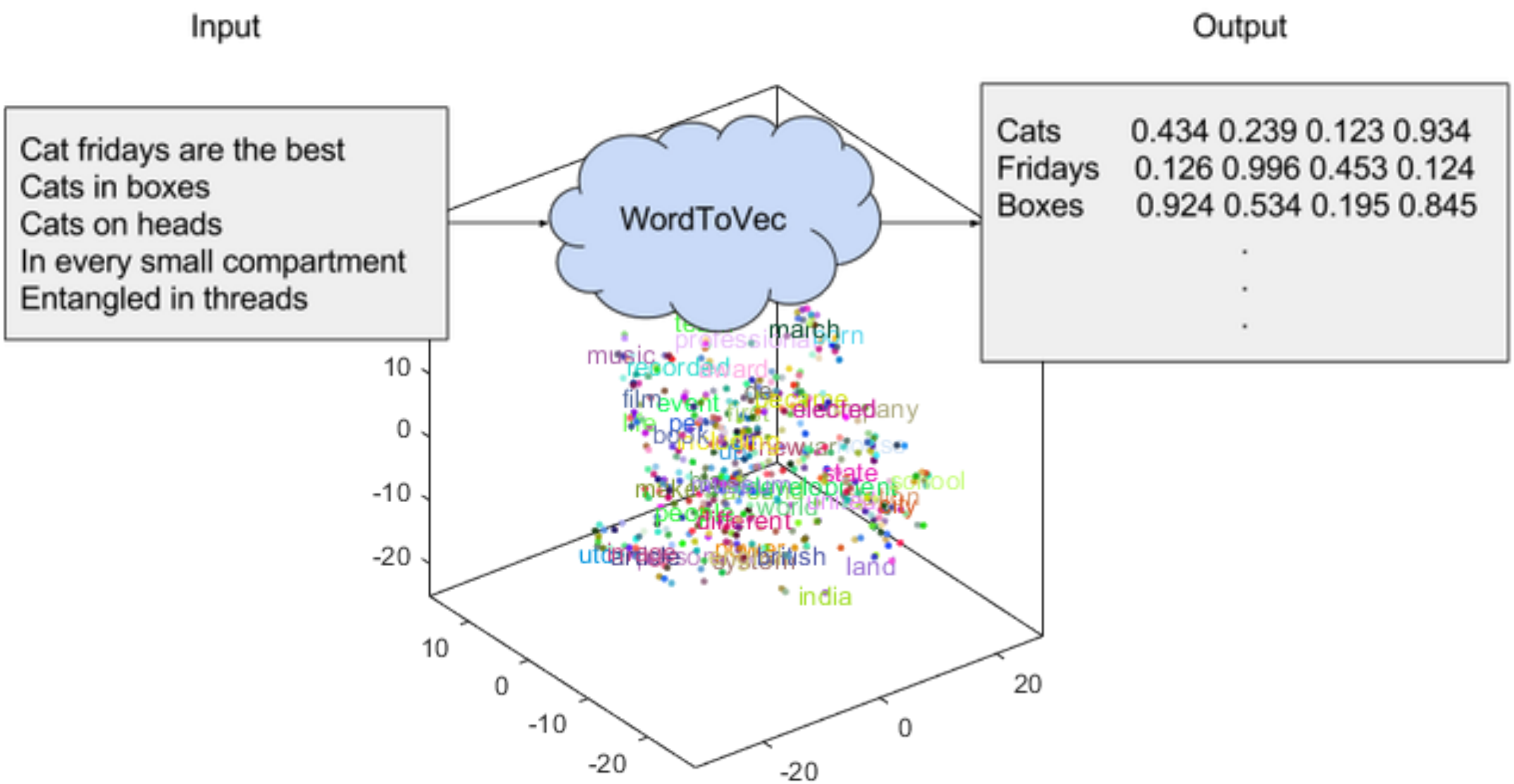
Machine Learning Lifecycle

- (Supervised) Machine Learning Lifecycle: This feature, that feature. Every single time!




Random Walk
word2vec
node2vec

word2vec


Input
Output
Cat fridays are the best
Cats in boxes
Cats on heads
In every small compartment
Entangled in threads
WordToVec
Cats 0.434 0.239 0.123 0.934
Fridays 0.126 0.996 0.453 0.124
Boxes 0.924 0.534 0.195 0.845
.
.
.
march
music
film
elected
state
school
different
british
land
india
10
0
-10
-20
10
0
-10
-20
20
0
-20

word2vec's backbone

$$J(\theta) = \frac{1}{T} \sum_{t=1}^{T} \sum_{-m \leq j \leq m, j \neq 0} \log p(w_{t+j} | w_t)$$

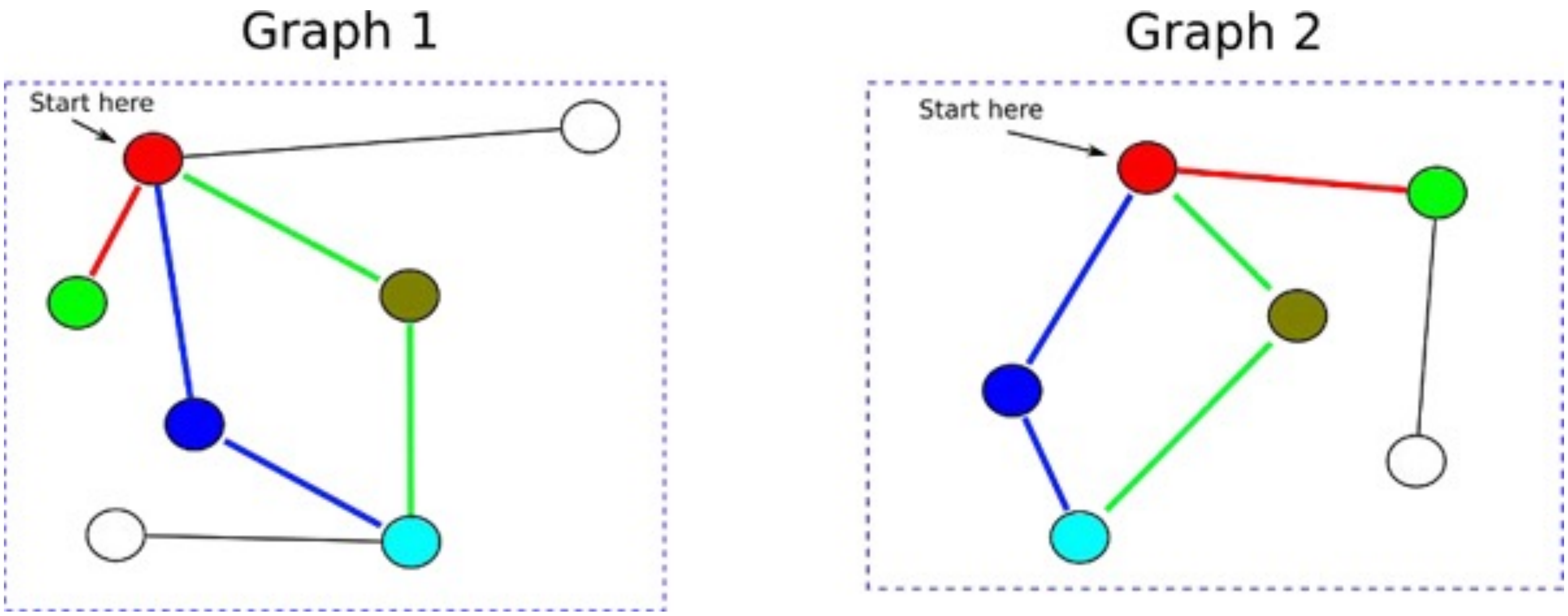
$$p(o|c) = \frac{\exp\left(u_o^T v_c\right)}{\sum_{w=1}^{W} \exp\left(u_w^T v_c\right)}$$

Window in Graph

Random Walk

Stochastic Process

Path of random steps



Feature Learning in Graphs

Goal: Learn features for a set of objects

Feature learning in graphs:

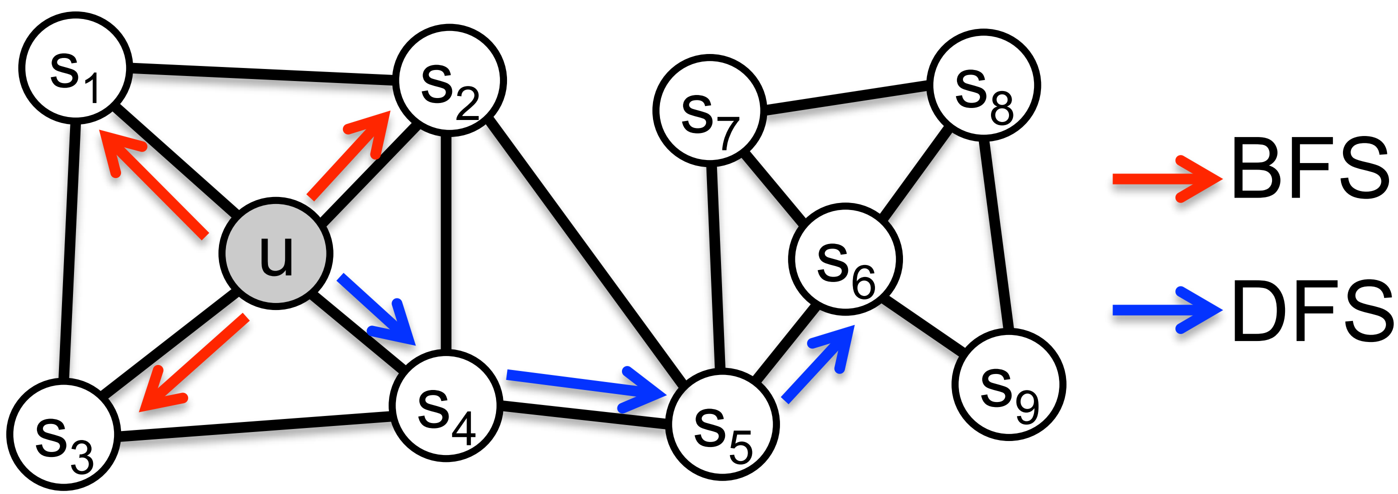
- Given: $G = (V, E)$
- Learn a function: $f : V \to \mathbb{R}^d$
  - Not task specific: Just given a graph, learn *f*. Can use the features for <u>any</u> downstream task!

Unsupervised Feature Learning

- Intuition: Find a mapping of nodes to d-dimensions that preserves some sort of node similarity
- Idea: Learn node embedding such that nearby nodes are close together
- Given a node *u*, how do we define nearby nodes?
  - $N_S(u)$ … neighbourhood of *u* obtained by sampling strategy *S*

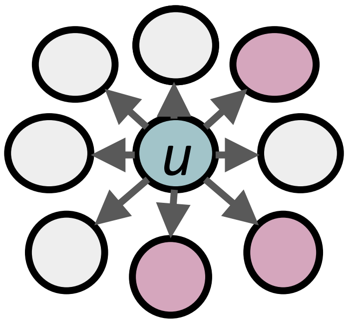
How to determine $N_S(u)$

Two classic search strategies to define a neighborhood of a given node:

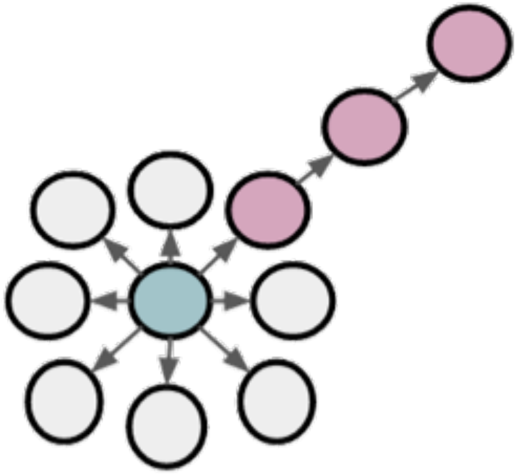


for $|N_S(u)| = 3$

BFS vs. DFS



BFS:
Micro-view of neighbourhood

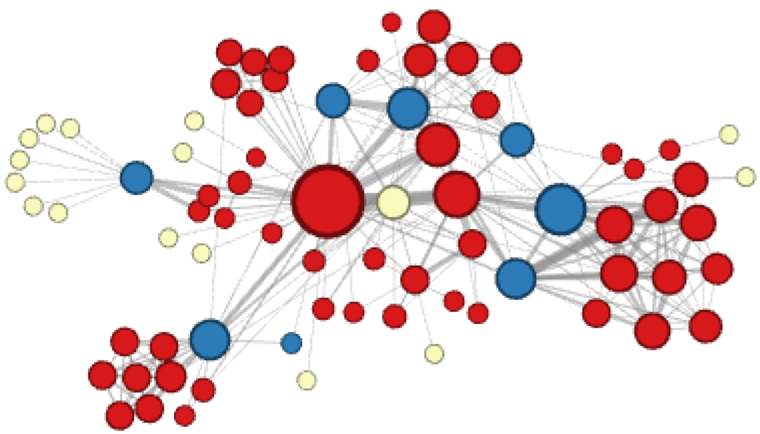


DFS:
Macro-view of neighbourhood

Structural vs. Homophilic equivalence

BFS vs. DFS

Structural vs. Homophilic equivalence

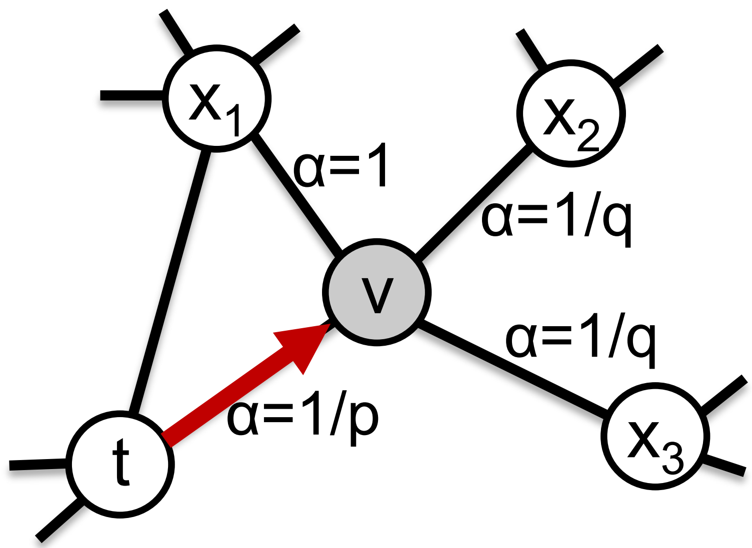


BFS-based:
Structural equivalence
(structural roles)

DFS-based:
Homophily
(network communities)

Interpolating BFS and DFS

- Biased random walk procedure, that given a node u samples $N_S(u)$



The walk just traversed (t, v) and aims to make a next step.

Multilabel Classification

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**

- Spectral embedding
- DeepWalk [B. Perozzi et al., KDD '14]
- LINE [J. Tang et al.. WWW '15]

Trade-offs

task-specific heuristics
inefficient usage of statistics

Thank you