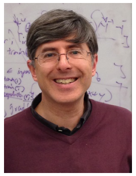
DeepWalk: Online Learning of Social Representations

ACM SIG-KDD
August 26, 2014



Bryan Perozzi, Rami Al-Rfou, Steven Skiena
Stony Brook University

Stony Brook University

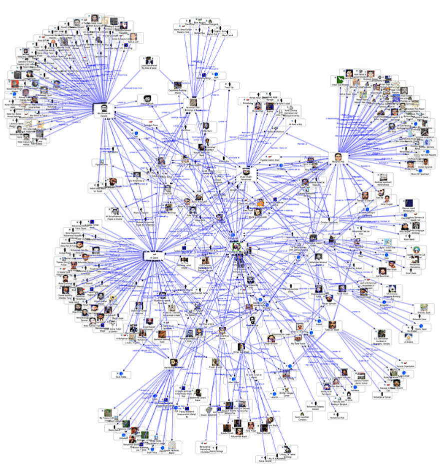
Outline

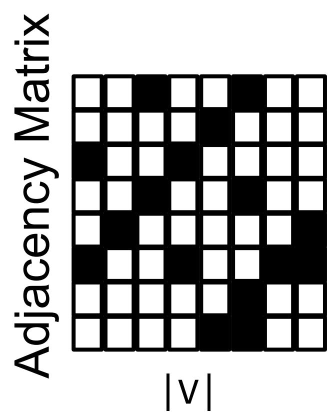
- **Introduction: Graphs as Features**
- Language Modeling
- DeepWalk
- Evaluation: Network Classification
- Conclusions & Future Work

Features From Graphs

A first step in machine learning for graphs is to extract graph features:

- node: degree
- pairs: # of common neighbors
- groups: cluster assignments

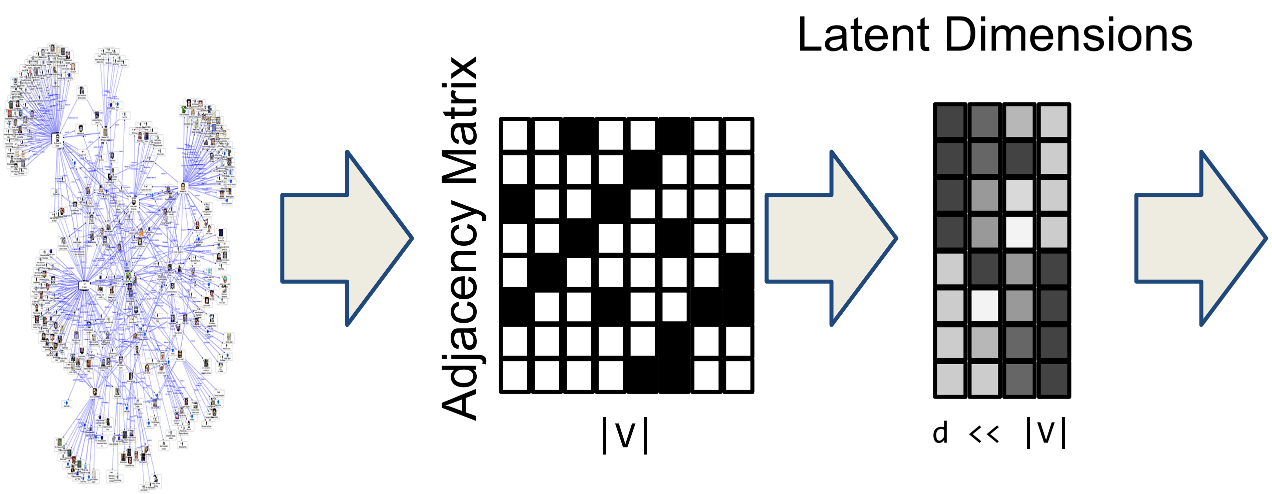




- Anomaly Detection
- Attribute Prediction
- Clustering
- Link Prediction
- ...

What is a Graph Representation?

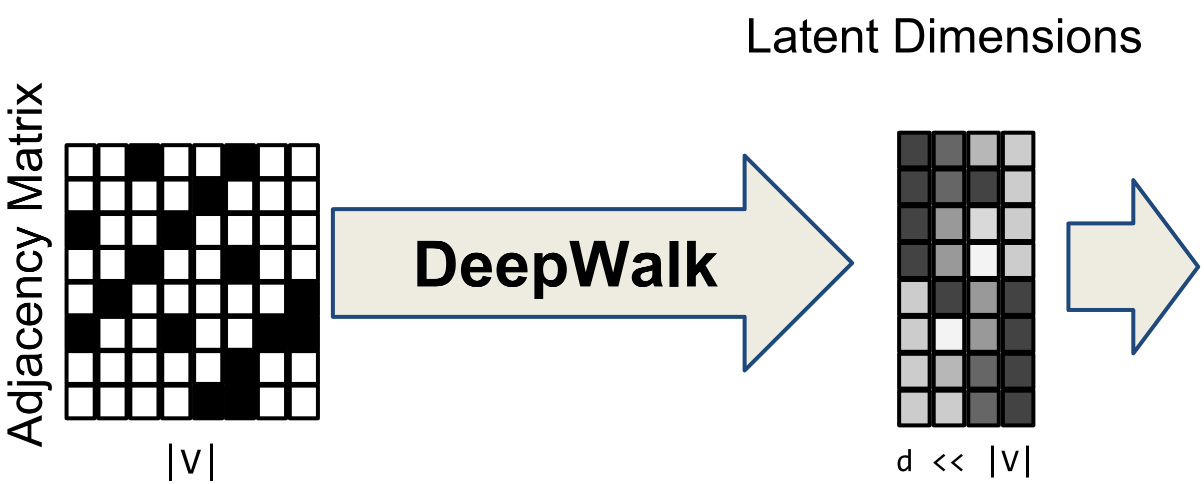
We can also create features by transforming the graph into a lower dimensional latent representation.



- Anomaly Detection
- Attribute Prediction
- Clustering
- Link Prediction
- ...

DeepWalk

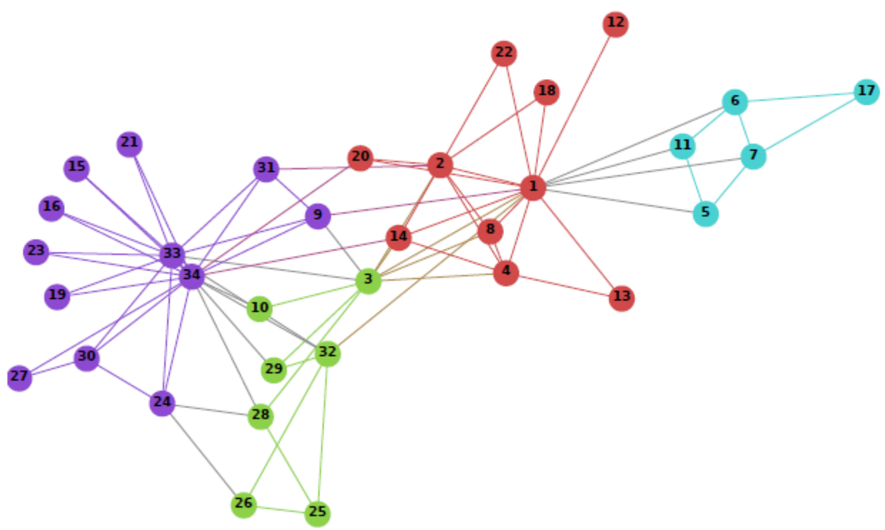
DeepWalk **learns** a latent representation of adjacency matrices using deep learning techniques developed for language modeling.

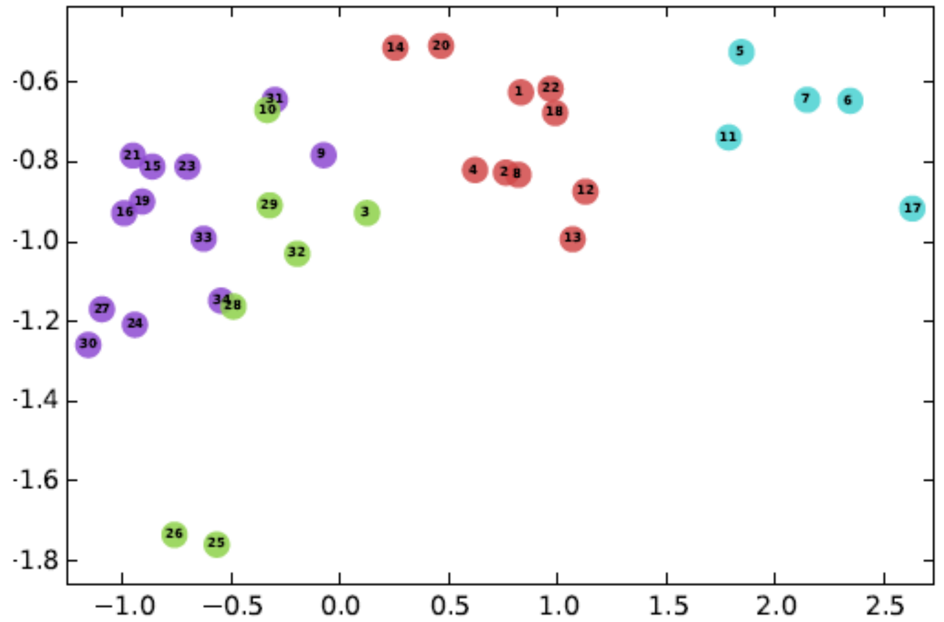


- Anomaly Detection
- Attribute Prediction
- Clustering
- Link Prediction
- ...

Visual Example

On Zachary's Karate Graph:



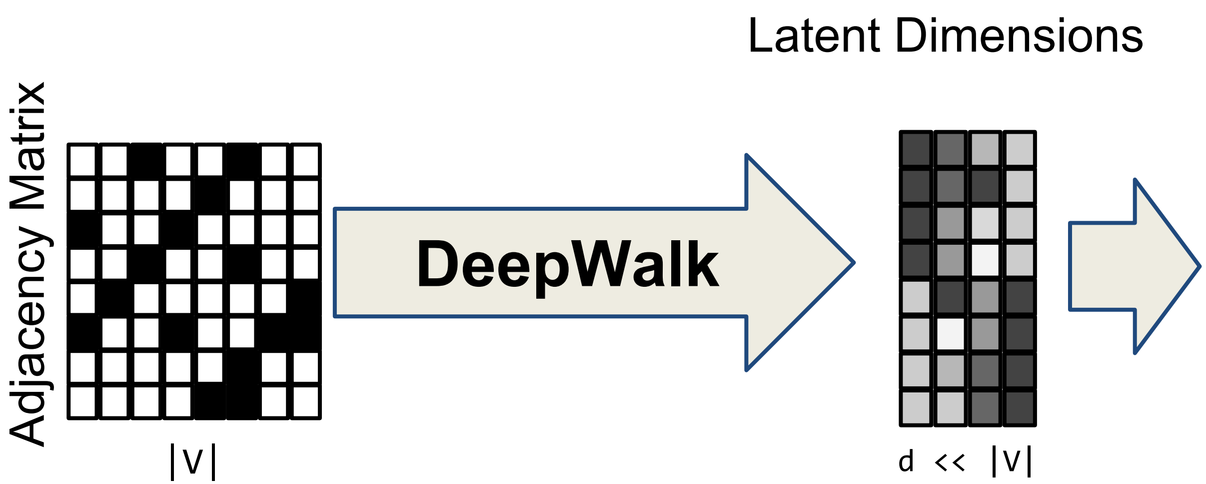


Input

Output

Advantages of DeepWalk

- Scalable - An online algorithm that does not use entire graph at once
- Walks as sentences metaphor
- Works great!
- Implementation available: bit.ly/deepwalk



- Anomaly Detection
- Attribute Prediction
- Clustering
- Link Prediction
- ...

Outline

- Introduction: Graphs as Features
- **Language Modeling**
- DeepWalk
- Evaluation: Network Classification
- Conclusions & Future Work

Language Modeling

Learning a representation means learning a mapping function from word co-occurrence

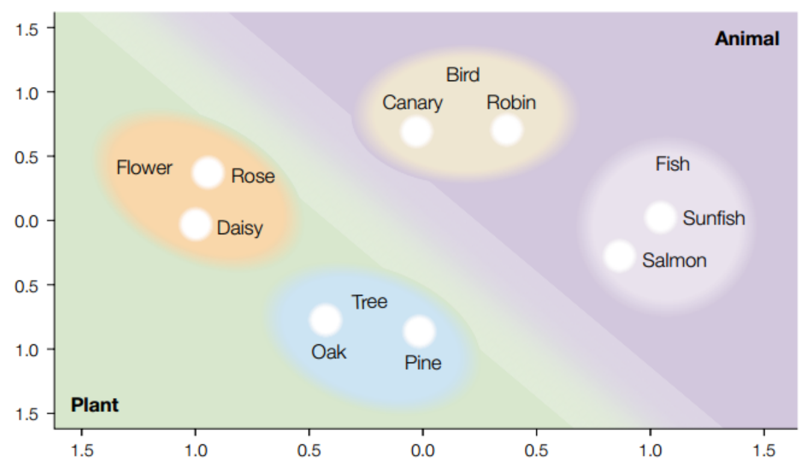
$$\Phi : v \in V \mapsto \mathbb{R}^{|V| \times d}$$

We hope that the learned representations capture inherent structure

$$||\Phi(rose) - \Phi(daisy)|| < ||\Phi(rose) - \Phi(tiger)||$$

```
rtains open and the moon shining in on the
nd the cold , close moon " . And neither o
 the night with the moon shining so bright
in the light of the moon . It all boils do
ly under a crescent moon , thrilled by ice
 the seasons of the moon ? Home , alone ,
dazzling snow , the moon has risen full an
d the temple of the moon , driving out of
```

[Baroni et al, 2009]



[Rumelhart+, 2003]

World of Word Embeddings

This is a very active research topic in NLP.

- **Importance sampling** and **hierarchical classification** were proposed to speed up training. [F. Morin and Y.Bengio, AISTATS 2005] [Y. Bengio and J. Sencal, IEEENN 2008] [A. Mnih, G. Hinton, NIPS 2008]

- **NLP applications** based on learned representations. [Colbert et al. **NLP (Almost) from Scratch**, (JMLR), 2011.]

- **Recurrent networks** were proposed to learn sequential representations. [Tomas Mikolov et al. ICASSP 2011]

- Composed representations learned through **recursive networks** were used for parsing, paraphrase detection, and sentiment analysis. [R. Socher, C. Manning, A. Ng, EMNLP (2011, 2012, 2013) NIPS (2011, 2012) ACL (2012, 2013)]

- **Vector spaces** of representations are developed to simplify **compositionality**. [T. Mikolov, G. Corrado, K. Chen and J. Dean, ICLR 2013, NIPS 2013]

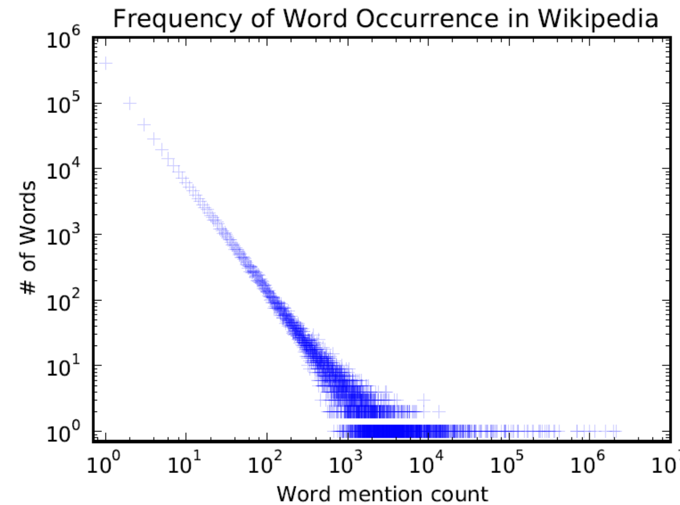
Word Frequency in Natural Language

```
rtains open and the moon shining in on the
nd the cold , close moon " . And neither o
 the night with the moon shining so bright
in the light of the moon . It all boils do
ly under a crescent moon , thrilled by ice
 the seasons of the moon ? Home , alone ,
dazzling snow , the moon has risen full an
d the temple of the moon , driving out of
```

Co-Occurrence Matrix

	planet	night	full	shadow	shine	crescent
moon	10	22	43	16	29	12
sun	14	10	4	15	45	0
dog	0	4	2	10	0	0

- Words frequency in a natural language corpus follows a power law.



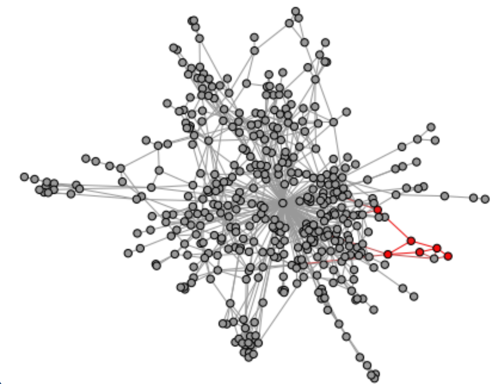
(b) Wikipedia Article Text

Connection: Power Laws

Vertex frequency in random walks on scale free graphs also follows a ***power law***.

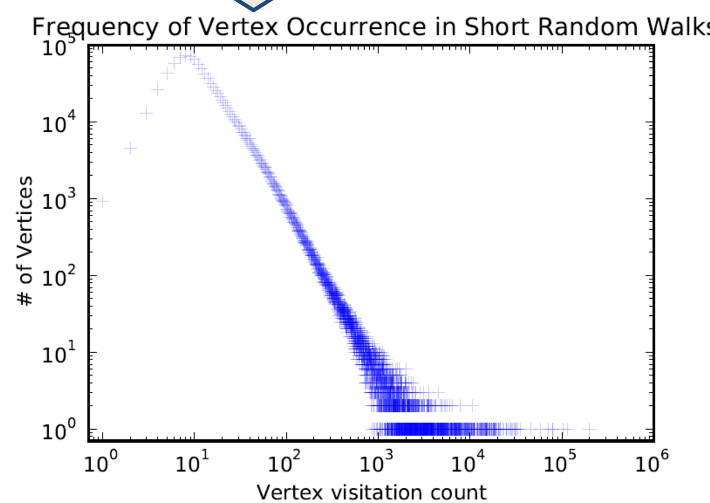
Vertex Frequency in SFG

$$
\begin{array}{llllll}
v_{71} \rightarrow & v_{24} \rightarrow & v_{5} \rightarrow & \color{red}{v_{1}} \rightarrow & v_{17} \rightarrow & v_{80} \rightarrow \\
v_{92} \rightarrow & v_{2} \rightarrow & v_{3} \rightarrow & \color{red}{v_{1}} \rightarrow & v_{12} \rightarrow & v_{73} \rightarrow \\
v_{37} \rightarrow & v_{34} \rightarrow & v_{9} \rightarrow & \color{red}{v_{1}} \rightarrow & v_{10} \rightarrow & v_{94} \rightarrow \\
v_{73} \rightarrow & v_{64} \rightarrow & v_{5} \rightarrow & \color{red}{v_{1}} \rightarrow & v_{12} \rightarrow & v_{1} \rightarrow \\
v_{75} \rightarrow & v_{14} \rightarrow & v_{6} \rightarrow & \color{red}{v_{1}} \rightarrow & v_{13} \rightarrow & v_{61} \rightarrow
\end{array}
$$



Scale Free Graph

- Short truncated random walks are sentences in an artificial language!
- Random walk distance is known to be good features for many problems



(a) YouTube Social Graph

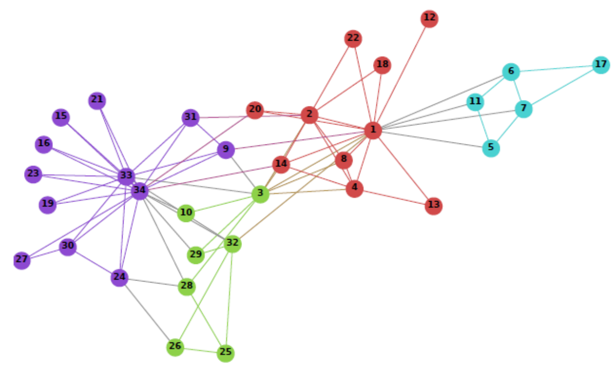
The Cool Idea

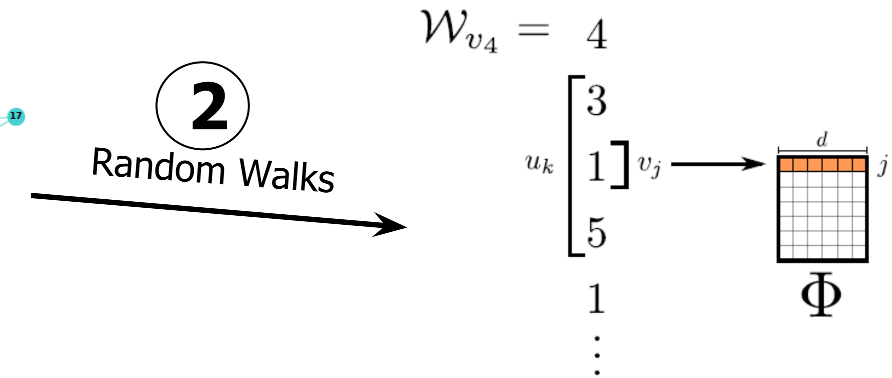
Short random walks = sentences

Outline

- Introduction: Graphs as Features
- Language Modeling
- **DeepWalk**
- Evaluation: Network Classification
- Conclusions & Future Work

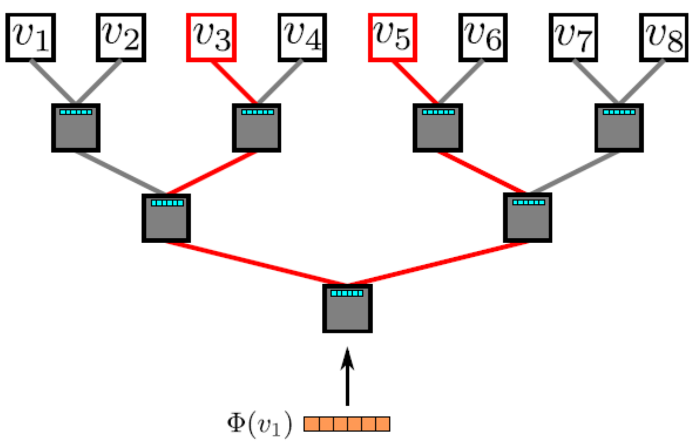
Deep Learning for Networks

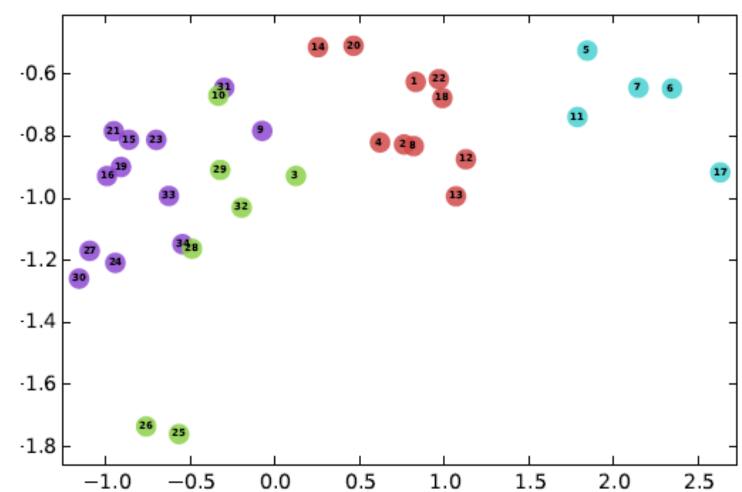




1 Input: Graph

3 Representation Mapping

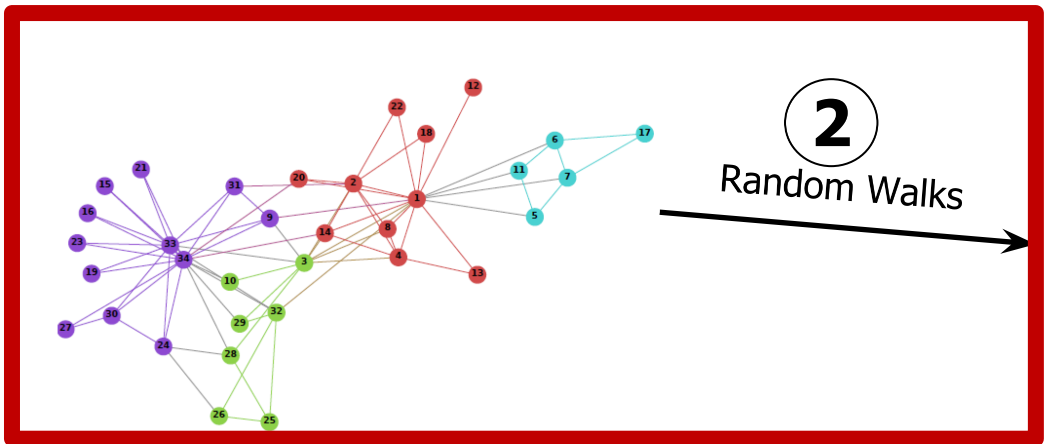


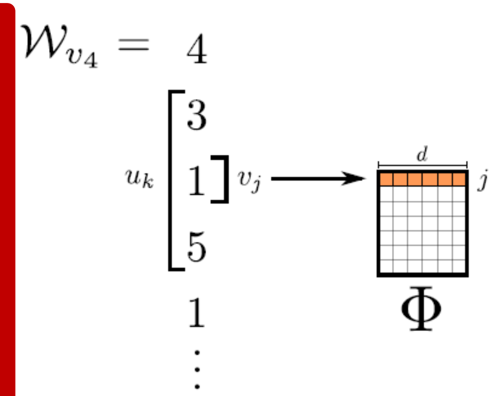


4 Hierarchical Softmax

5 Output: Representation

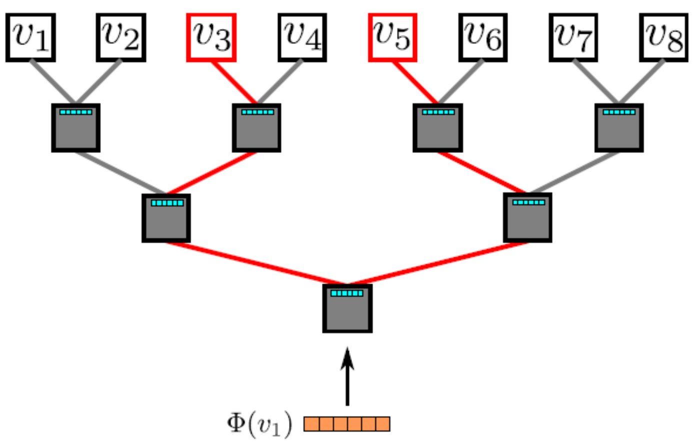
Deep Learning for Networks

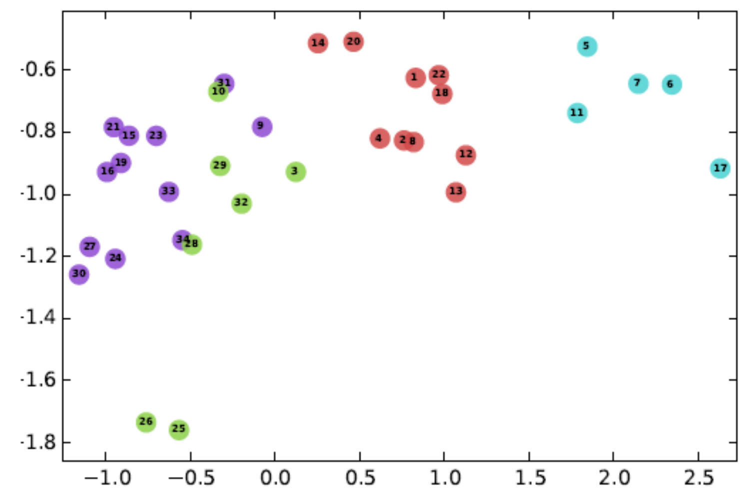




1 Input: Graph

3 Representation Mapping

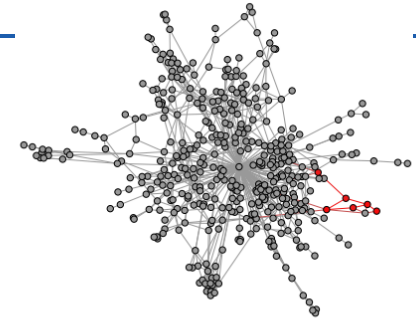




4 Hierarchical Softmax

5 Output: Representation

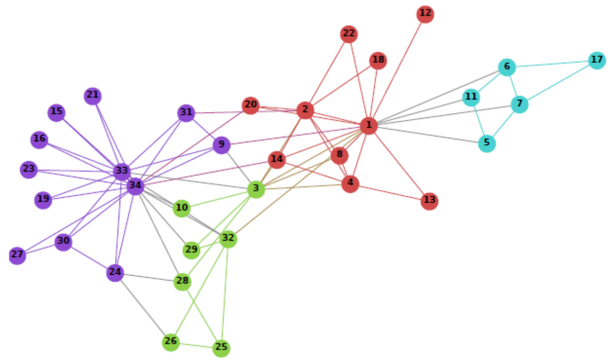
Random Walks



- We generate γ random walks for each vertex in the graph.
- Each short random walk has length t .
- Pick the next step ***uniformly*** from the vertex neighbors.
- Example:

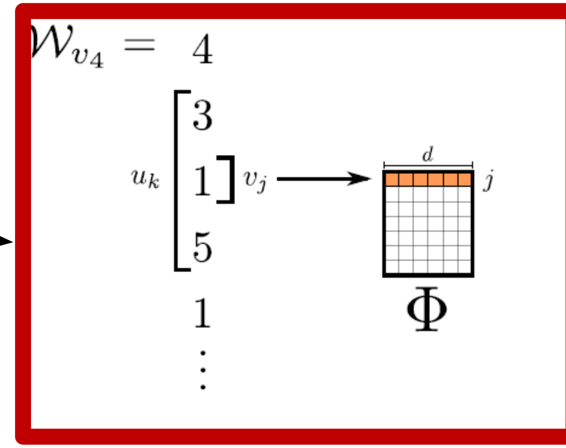
$$v_{46} \rightarrow v_{45} \rightarrow v_{71} \rightarrow v_{24} \rightarrow v_{5} \rightarrow {\color{red} v_{1}} \rightarrow v_{17}$$

Deep Learning for Networks

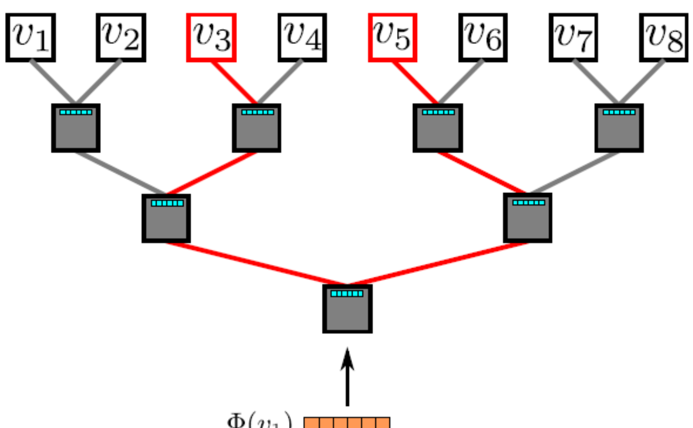


1 Input: Graph

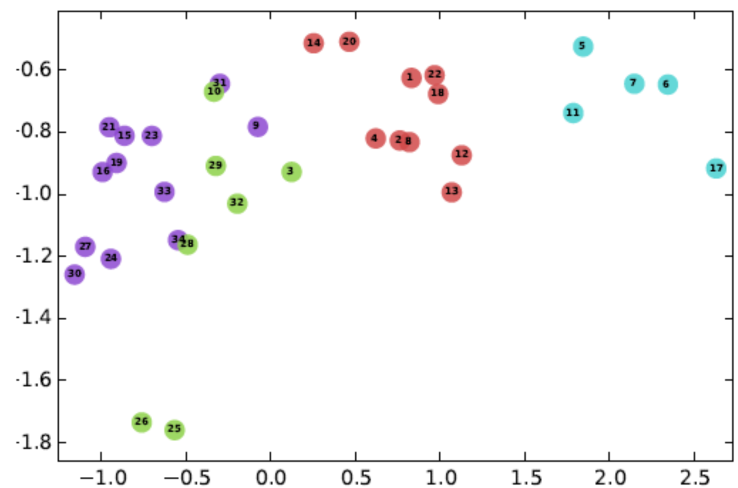
2 Random Walks



3 Representation Mapping



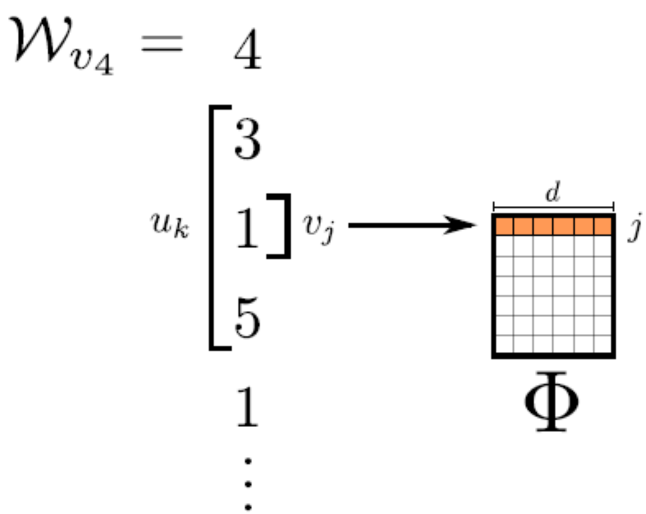
4 Hierarchical Softmax



5 Output: Representation

Representation Mapping

$$\mathcal{W}_{v_4} \equiv v_4 \rightarrow v_3 \rightarrow v_1 \rightarrow v_5 \rightarrow v_1 \rightarrow v_{46} \rightarrow v_{51} \rightarrow v_{89}$$

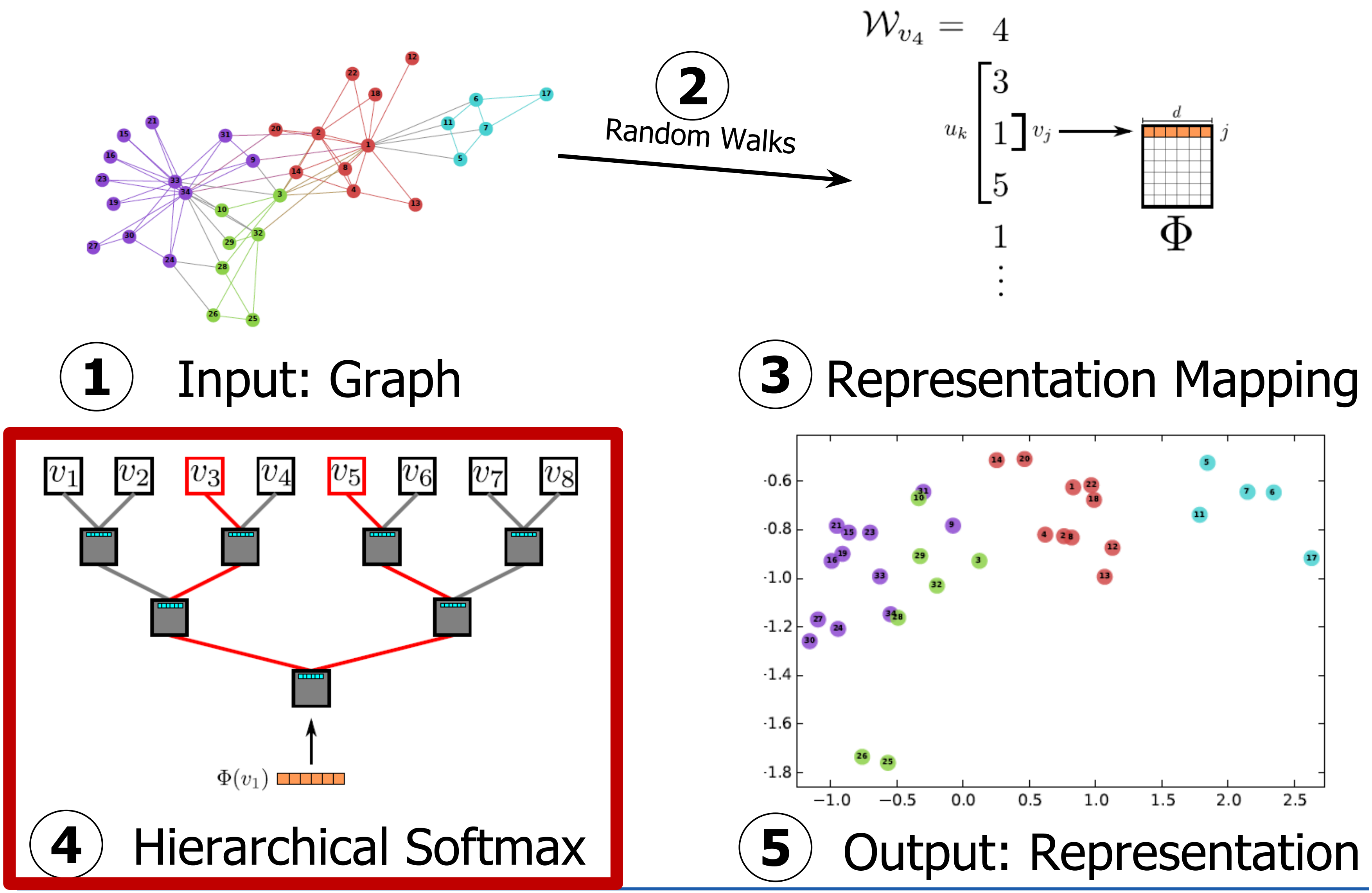


- Map the vertex under focus (v_1) to its representation.
- Define a window of size w
- If w = 1 and v = v_1

Maximize: $\Pr(v_3 | \Phi(v_1))$

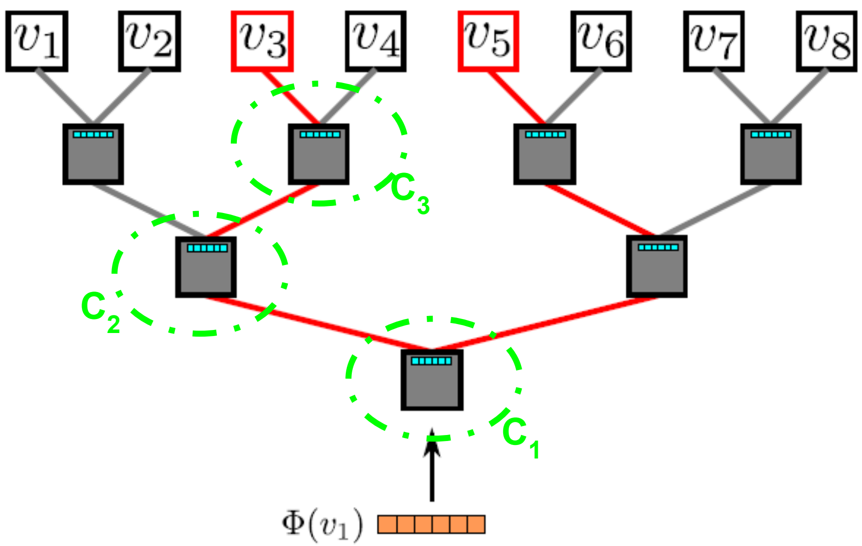
$\Pr(v_5 | \Phi(v_1))$

Deep Learning for Networks



Hierarchical Softmax

Calculating $\Pr(v_3|\Phi(v_1))$ involves O(V) operations for each update! Instead:



- Consider the graph vertices as leaves of a balanced binary tree.
- **Maximizing** $\Pr(v_3|\Phi(v_1))$ is equivalent to maximizing the probability of the path from the root to the node. specifically, maximizing

$$\Pr(right \mid \Phi(v_1); C_2)$$

$$\Pr(left \mid \Phi(v_1); C_3)$$

$$\Pr(left \mid \Phi(v_1); C_1)$$

Each of {C_1, C_2, C_3} is a logistic binary classifier.

Learning

[Mikolov+, 2013]

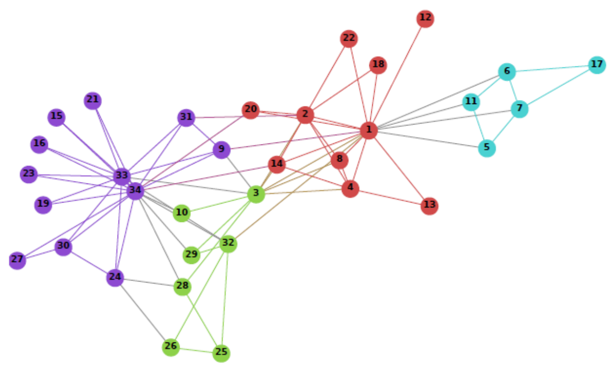
- Learned parameters:
  - Vertex representations
  - Tree binary classifiers weights

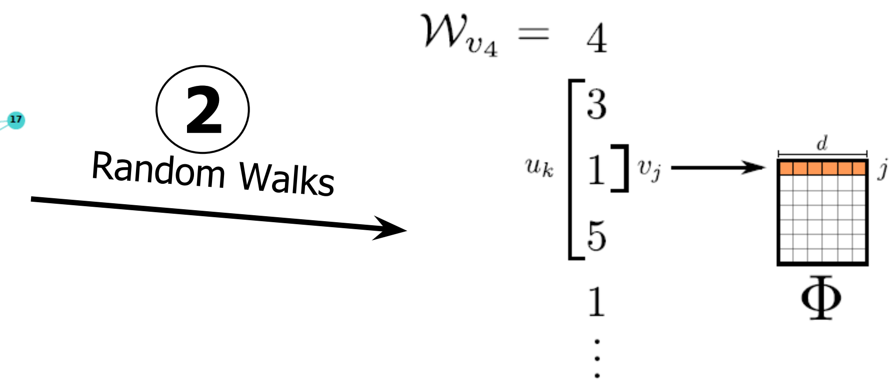
- Randomly initialize the representations.

- For each {C_1, C_2, C_3} calculate the loss function.

- Use Stochastic Gradient Descent to update both the *classifier weights* and the *vertex representation* ***simultaneously***.

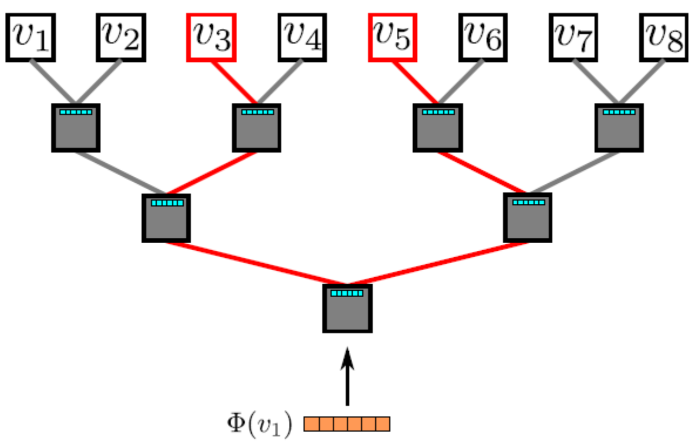
Deep Learning for Networks

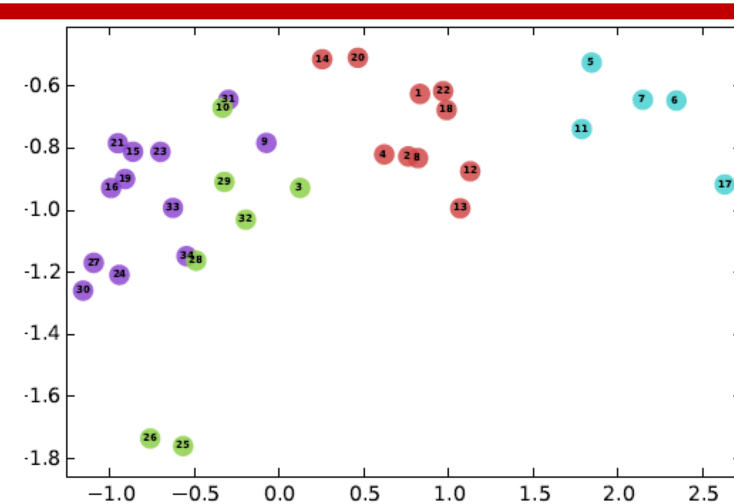




1 Input: Graph

3 Representation Mapping





4 Hierarchical Softmax

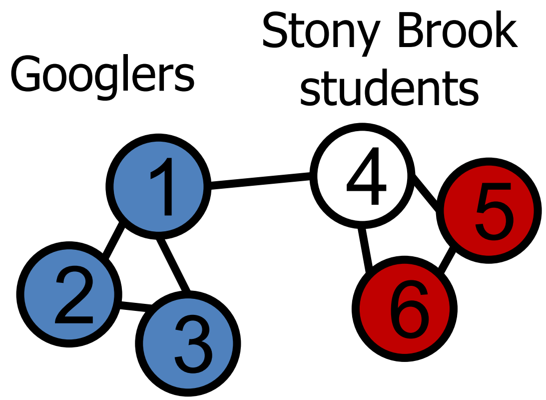
5 Output: Representation

Outline

- Introduction: Graphs as Features
- Language Modeling
- DeepWalk
- **Evaluation: Network Classification**
- Conclusions & Future Work

Attribute Prediction

The Semi-Supervised Network Classification problem:

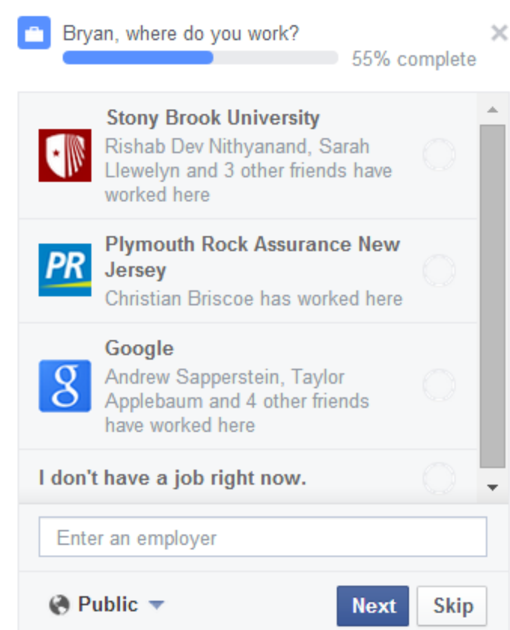


INPUT

A partially labelled graph with node attributes.

OUTPUT

Attributes for nodes which do not have them.



Baselines

- Approximate Inference Techniques:
  - weighted vote Relational Neighbor ($\mathtt{wvRN}$)[Macskassy+, '03]
- Latent Dimensions
  - Spectral Methods
    - SpectralClustering [Tang+, '11]
    - MaxModularity [Tang+, '09]
  - k-means
    - EdgeCluster [Tang+, '09]

Results: BlogCatalog

Name	BlogCatalog
$\|V\|$	10,312
$\|E\|$	333,983
$\|\mathcal{Y}\|$	39
Labels	Interests

	% Labeled Nodes	10%	20%	30%	40%	50%	60%	70%	80%	90%
Micro-F1(%)	DeepWalk	**36.00**	**38.20**	**39.60**	**40.30**	**41.00**	**41.30**	41.50	41.50	42.00
	SpectralClustering	31.06	34.95	37.27	38.93	39.97	40.99	**41.66**	**42.42**	**42.62**
	EdgeCluster	27.94	30.76	31.85	32.99	34.12	35.00	34.63	35.99	36.29
	Modularity	27.35	30.74	31.77	32.97	34.09	36.13	36.08	37.23	38.18
	wvRN	19.51	24.34	25.62	28.82	30.37	31.81	32.19	33.33	34.28
	Majority	16.51	16.66	16.61	16.70	16.91	16.99	16.92	16.49	17.26
Macro-F1(%)	DeepWalk	**21.30**	**23.80**	25.30	26.30	27.30	27.60	27.90	28.20	28.90
	SpectralClustering	19.14	23.57	**25.97**	**27.46**	**28.31**	**29.46**	**30.13**	**31.38**	**31.78**
	EdgeCluster	16.16	19.16	20.48	22.00	23.00	23.64	23.82	24.61	24.92
	Modularity	17.36	20.00	20.80	21.85	22.65	23.41	23.89	24.20	24.97
	wvRN	6.25	10.13	11.64	14.24	15.86	17.18	17.98	18.86	19.57
	Majority	2.52	2.55	2.52	2.58	2.58	2.63	2.61	2.48	2.62

Table 2: Multi-label classification results in BlogCatalog

DeepWalk performs well, especially when labels are sparse.

Results: Flickr

Name	FLICKR
$\lvert V \rvert$	80,513
$\lvert E \rvert$	5,899,882
$\lvert \mathcal{Y} \rvert$	195
Labels	Groups

	% Labeled Nodes	1%	2%	3%	4%	5%	6%	7%	8%	9%	10%
Micro-F1(%)	DEEPWALK	**32.4**	**34.6**	**35.9**	**36.7**	**37.2**	**37.7**	**38.1**	**38.3**	**38.5**	**38.7**
	SpectralClustering	27.43	30.11	31.63	32.69	33.31	33.95	34.46	34.81	35.14	35.41
	EdgeCluster	25.75	28.53	29.14	30.31	30.85	31.53	31.75	31.76	32.19	32.84
	Modularity	22.75	25.29	27.3	27.6	28.05	29.33	29.43	28.89	29.17	29.2
	wvRN	17.7	14.43	15.72	20.97	19.83	19.42	19.22	21.25	22.51	22.73
	Majority	16.34	16.31	16.34	16.46	16.65	16.44	16.38	16.62	16.67	16.71
Macro-F1(%)	DEEPWALK	**14.0**	17.3	**19.6**	**21.1**	**22.1**	**22.9**	**23.6**	**24.1**	**24.6**	**25.0**
	SpectralClustering	13.84	**17.49**	19.44	20.75	21.60	22.36	23.01	23.36	23.82	24.05
	EdgeCluster	10.52	14.10	15.91	16.72	18.01	18.54	19.54	20.18	20.78	20.85
	Modularity	10.21	13.37	15.24	15.11	16.14	16.64	17.02	17.1	17.14	17.12
	wvRN	1.53	2.46	2.91	3.47	4.95	5.56	5.82	6.59	8.00	7.26
	Majority	0.45	0.44	0.45	0.46	0.47	0.44	0.45	0.47	0.47	0.47

Table: Multi-label classification results in FLICKR

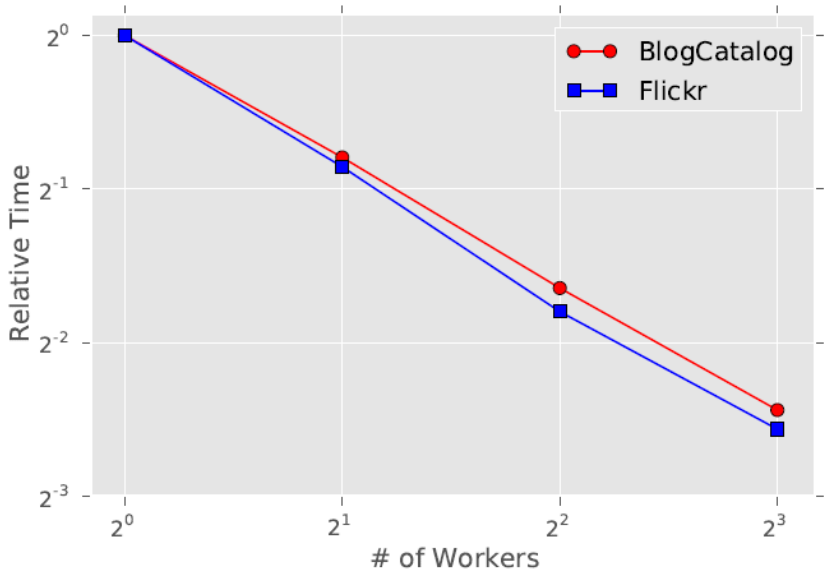
Results: YouTube

Name	YouTube
$\|V\|$	1,138,499
$\|E\|$	2,990,443
$\|\mathcal{Y}\|$	47
Labels	Groups

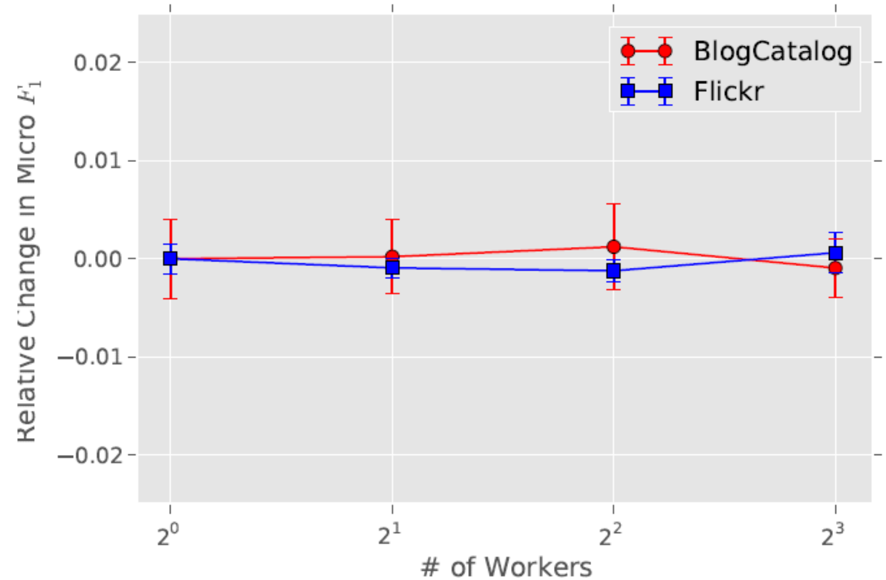
	% Labeled Nodes	1%	2%	3%	4%	5%	6%	7%	8%	9%	10%
Micro-F1(%)	DeepWalk	**37.95**	**39.28**	**40.08**	**40.78**	**41.32**	**41.72**	**42.12**	**42.48**	**42.78**	**43.05**
	SpectralClustering	—	—	—	—	—	—	—	—	—	—
	EdgeCluster	23.90	31.68	35.53	36.76	37.81	38.63	38.94	39.46	39.92	40.07
	Modularity	—	—	—	—	—	—	—	—	—	—
	wvRN	26.79	29.18	33.1	32.88	35.76	37.38	38.21	37.75	38.68	39.42
	Majority	24.90	24.84	25.25	25.23	25.22	25.33	25.31	25.34	25.38	25.38
Macro-F1(%)	DeepWalk	**29.22**	**31.83**	**33.06**	**33.90**	**34.35**	**34.66**	**34.96**	**35.22**	**35.42**	**35.67**
	SpectralClustering	—	—	—	—	—	—	—	—	—	—
	EdgeCluster	19.48	25.01	28.15	29.17	29.82	30.65	30.75	31.23	31.45	31.54
	Modularity	—	—	—	—	—	—	—	—	—	—
	wvRN	13.15	15.78	19.66	20.9	23.31	25.43	27.08	26.48	28.33	28.89
	Majority	6.12	5.86	6.21	6.1	6.07	6.19	6.17	6.16	6.18	6.19

Spectral Methods do not scale to large graphs.

Parallelization



(a) Running Time



(b) Performance

- Parallelization doesn’t affect representation quality.
- The sparser the graph, the easier to achieve linear scalability. (Feng+, NIPS ‘11)

Outline

- Introduction: Graphs as Features
- Language Modeling
- DeepWalk
- Evaluation: Network Classification
- **Conclusions & Future Work**

Variants / Future Work

- Streaming
  - No need to ever store entire graph
  - Can build & update representation as new data comes in.
- "Non-Random" Walks
  - Many graphs occur through as a by-product of interactions
  - One could outside processes (users, etc) to feed the modeling phase
  - [This is what language modeling is doing]

Take-away

Language Modeling techniques can be used for online learning of network representations.

Thanks!

Bryan Perozzi
@phanein

bperozzi@cs.stonybrook.edu

DeepWalk available at:
http://bit.ly/deepwalk