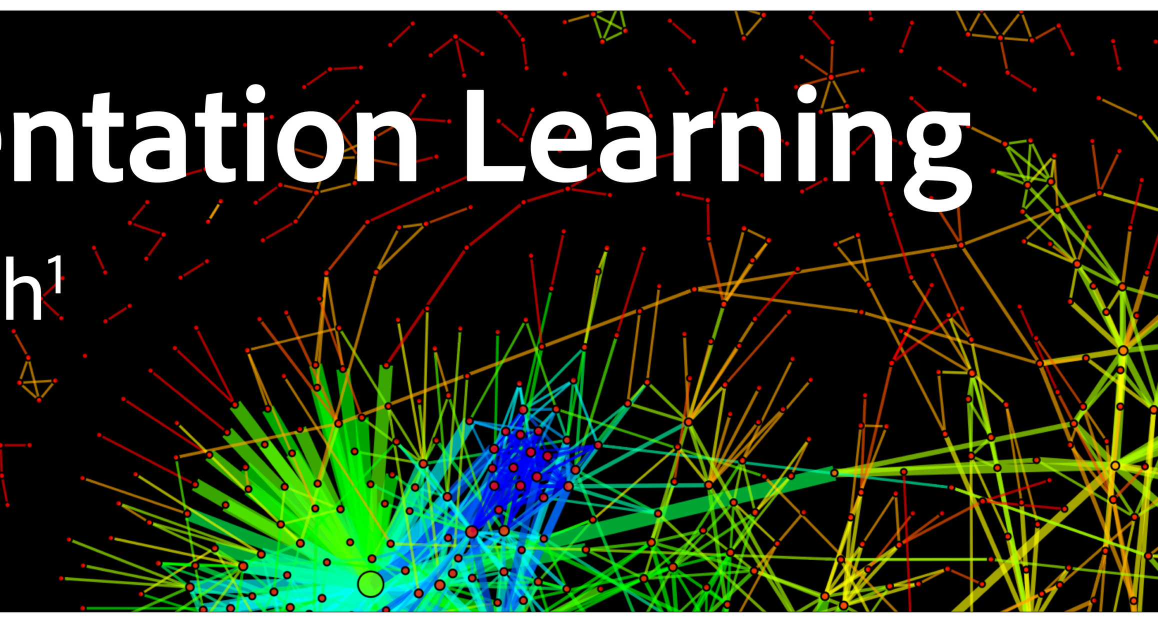
Higher-order Network Representation Learning

Ryan A. Rossi[1], Nesreen K. Ahmed[2], and Eunyee Koh[1]

[1]Adobe Research, San Jose, CA
[2]Intel Labs, Santa Clara, CA



Overview

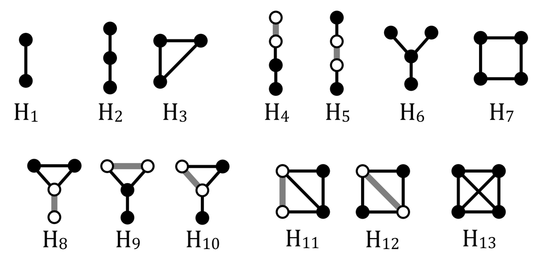
Learning a useful representation (embeddings) from graph data lies at the heart and success of many machine learning tasks such as entity resolution, classification, link prediction, and anomaly detection. However, existing embedding methods are unable to capture higher-order dependencies and connectivity patterns that are crucial to understanding and modeling complex networks.

In this work, we formulate higher-order network representation learning and describe a general framework for learning ***Higher-Order Network Embeddings*** (HONE) from graph data based on subgraph patterns called graphlets (network motifs, orbits). The HONE framework is highly expressive and flexible with many interchangeable components. The experimental results demonstrate the effectiveness of learning higher-order network representations. In all cases, HONE outperforms recent embedding methods that are unable to capture higher-order structures. In particular, HONE achieve a mean relative gain in AUC of 19% (and up to 75% gain) across all methods and over a wide variety of networks from different application domains.

Higher-Order Network Embeddings (HONE)

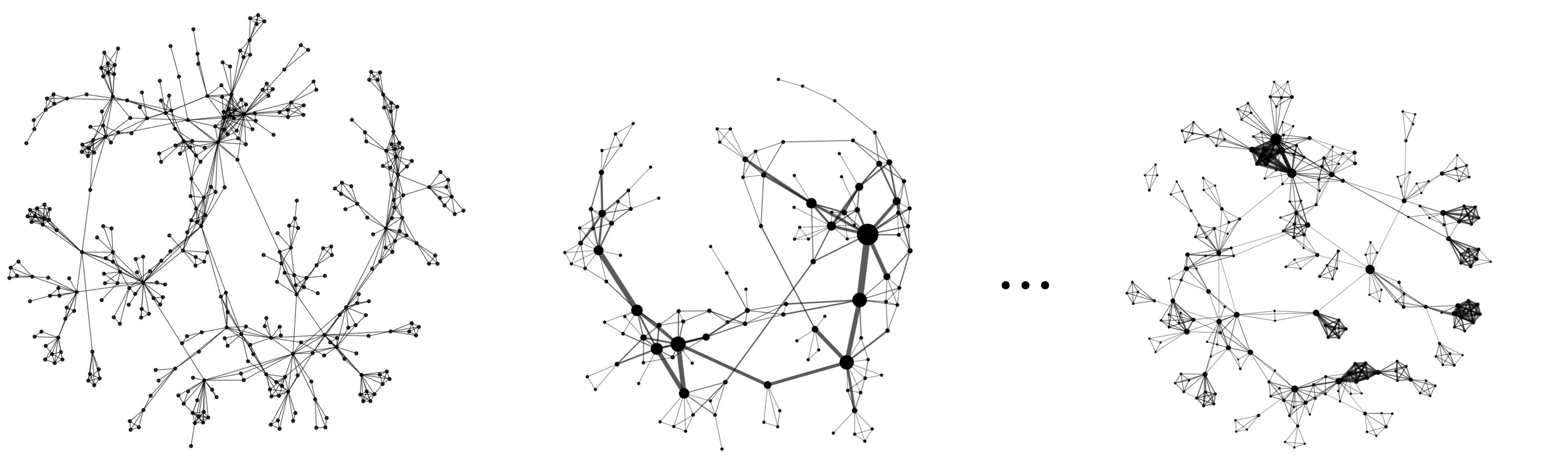
Given a graph G=(V,E) and a set $\mathcal{H} = \{H_1, \ldots, H_T\}$ of T network motifs, we form the weighted motif adjacency matrices

$$\mathcal{W} = \{\mathbf{W}_1, \mathbf{W}_2, \ldots, \mathbf{W}_T\}$$



$(\mathbf{W}_t)_{ij}$ = # of instances of motif H_t that contain nodes i and j (Eq. 2)

(thought of as a tensor where $\mathbf{W}_{ijt}$ = # instances of H_t that contain (i, j) ∈ E)



Initial graph — Weighted 4-path motif graph … Weighted 4-clique motif graph

Motif Matrix Formulations

To generalize HONE for any motif-based matrix formulation, we can define a function

$$\Psi : \mathbb{R}^{N \times N} \rightarrow \mathbb{R}^{N \times N}$$

Motif Transition Matrix:

$$\mathbf{P} = \mathbf{D}^{-1}\mathbf{W} \qquad \text{(Eq. 4)}$$
$$\mathbf{D} = \mathtt{diag}(\mathbf{We})$$
$$\textstyle\sum_j P_{ij} = \mathbf{p}_i^{\intercal}\mathbf{e} = 1$$

Motif Laplacian Matrix:

$$\mathbf{L} = \mathbf{D} - \mathbf{W} \qquad \text{(Eq. 5)}$$

(can use in/out/total motif degree)

Normalized Motif Laplacian:

$$\widehat{\mathbf{L}} = \mathbf{I} - \mathbf{D}^{-1/2}\mathbf{W}\mathbf{D}^{-1/2} \qquad \text{(Eq. 6)}$$

$$\widehat{L}_{ij} = \begin{cases} 1 - \frac{W_{ij}}{w_j} & \text{if } i = j \text{ and } w_j \neq 0 \\ -\frac{W_{ij}}{\sqrt{w_i w_j}} & \text{if } i \text{ and } j \text{ are adjacent} \\ 0 & \text{otherwise} \end{cases}$$

$w_i = \sum_j W_{ij}$ is the motif degree of node i

Other interesting motif matrix formulations can also be used!

K-step Motif-based Matrices

Given the motif matrix function Ψ and the weighted motif graphs $\mathcal{W}$, we derive all k-step motif-based matrices for all T motifs and K steps

$$\mathbf{S}_t^{(k)} = \Psi(\mathbf{W}_t^k), \quad \text{for } k = 1, \ldots, K \text{ and } t = 1, \ldots, T$$

The number of paths weighted by motif counts from node i to node j in k-steps is

$$(\mathbf{W}^k)_{ij} = (\underbrace{\mathbf{W} \cdots \mathbf{W}}_{k})_{ij}$$

The probability of transitioning from node i to node j in k-steps is given by

$$(\mathbf{P}^k)_{ij} = (\underbrace{\mathbf{P} \cdots \mathbf{P}}_{k})_{ij}$$

Local & Global Embeddings

We find low-rank "local" node embeddings for each motif and k-step matrix by solving the following optimization problem:

$$\underset{\mathbf{U}_t^{(k)}, \mathbf{V}_t^{(k)} \in \mathcal{C}}{\arg\min} \; \mathbb{D}\big(\mathbf{S}_t^{(k)} \,\|\, \Phi\langle \mathbf{U}_t^{(k)} \mathbf{V}_t^{(k)} \rangle\big), \quad \text{for k=1,...,K and t=1,...,T}$$

$$\mathbf{S}_t^{(k)} \approx \Phi\langle \mathbf{U}_t^{(k)} \mathbf{V}_t^{(k)} \rangle$$

Concatenate all k-step node embedding for all T motifs and K steps

$$\mathbf{Y} = \Big[\, \underbrace{\mathbf{U}_1^{(1)} \cdots \mathbf{U}_T^{(1)}}_{\text{1-step}} \;\cdots\; \underbrace{\mathbf{U}_1^{(K)} \cdots \mathbf{U}_T^{(K)}}_{K\text{-steps}} \,\Big] \qquad \mathbf{Y} \text{ is } N \times TKD_\ell$$

All node embeddings are normalized appropriately via L2 norm

$$\mathbf{U}_t^{(k)} \leftarrow g(\mathbf{U}_t^{(k)}), \quad \text{for } t = 1, \ldots, T \text{ and } k = 1, \ldots, K$$

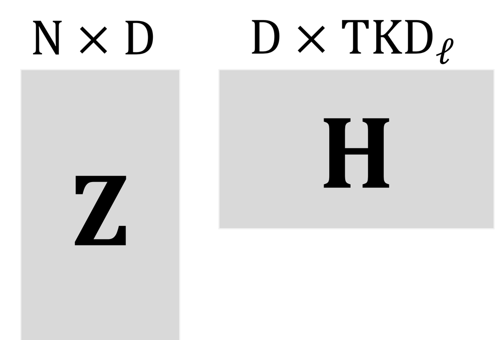
and find a "global" higher-order node embeddings by solving

$$\underset{\mathbf{Z}, \mathbf{H} \in \mathcal{C}}{\arg\min} \; \mathbb{D}\big(\mathbf{Y} \,\|\, \Phi\langle \mathbf{ZH} \rangle\big) \quad \text{(general Bregman divergence)}$$

For instance,

$$\min_{\mathbf{Z},\mathbf{H}} \frac{1}{2} \|\mathbf{Y} - \mathbf{ZH}\|_F^2 = \frac{1}{2} \sum_{ij} \big(\mathbf{Y}_{ij} - (\mathbf{ZH})_{ij}\big)^2$$

- Each row of **Z** is a D-dimensional embedding of a node



Evaluation & Results

Experimental setup: 10-fold cross-validation, repeated for 10 random trials, D=128, D_l = 16, edge embedding derived via $(\mathbf{z}_i + \mathbf{z}_j)/2$; Predict link existence via LR; # steps K selected via grid search $K \in \{1, 2, 3, 4\}$

AUC results comparing HONE to recent embedding methods across a wide variety of networks from different application domains.

	soc-hamster	*rt-twitter-cop*	*soc-wiki-Vote*	*tech-routers-rf*	*facebook-PU*	*inf-openflights*	*soc-bitcoinA*	Rank
HONE-W (Eq. 2)	0.841	0.843	0.811	0.862	0.726	0.910	0.979	1
HONE-P (Eq. 4)	0.840	0.840	0.812	0.863	0.724	0.913	0.980	2
HONE-L (Eq. 5)	0.829	0.841	0.808	0.858	0.722	0.906	0.975	3
HONE-$\widehat{\text{L}}$ (Eq. 6)	0.829	0.836	0.803	0.862	0.722	0.908	0.976	4
Node2Vec [4]	0.810	0.635	0.721	0.804	0.701	0.844	0.894	5
DeepWalk [5]	0.796	0.621	0.710	0.796	0.696	0.837	0.863	6
LINE [9]	0.752	0.706	0.734	0.800	0.630	0.837	0.780	7
GraRep [3]	0.805	0.672	0.743	0.829	0.702	0.898	0.559	8
Spectral [10]	0.561	0.699	0.593	0.602	0.516	0.606	0.629	9

Overall improvement in AUC of 19.2% and up to 75.2%

Main Findings & Contributions

1. Introduced higher-order network embeddings (HONE)
2. Described a computational framework for computing them
3. Demonstrated the effectiveness of higher-order network embeddings as HONE achieves a mean relative gain in AUC of 19% across all other methods and networks from a wide variety of application domains.