Asymmetric Transitivity Preserving Graph Embedding

Mingshan

7 June

OUTLINE

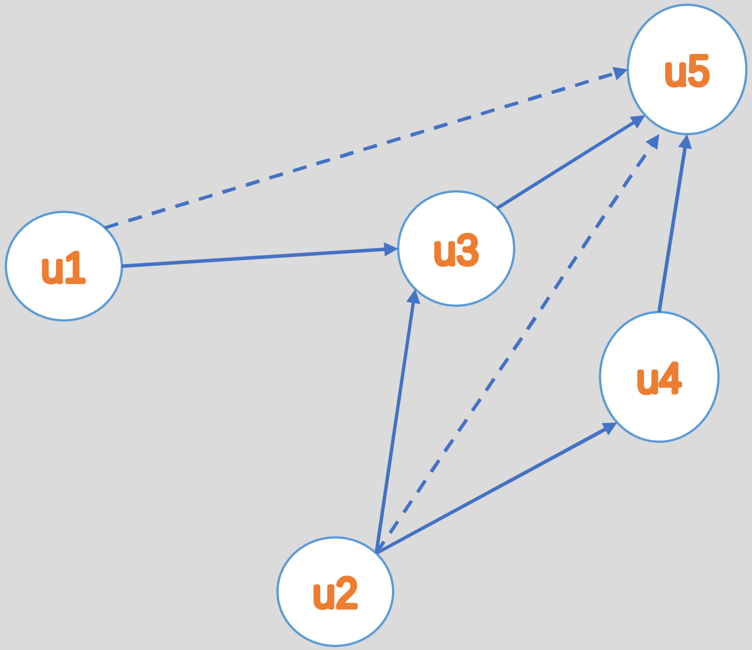
GRAPH EMBEDDING

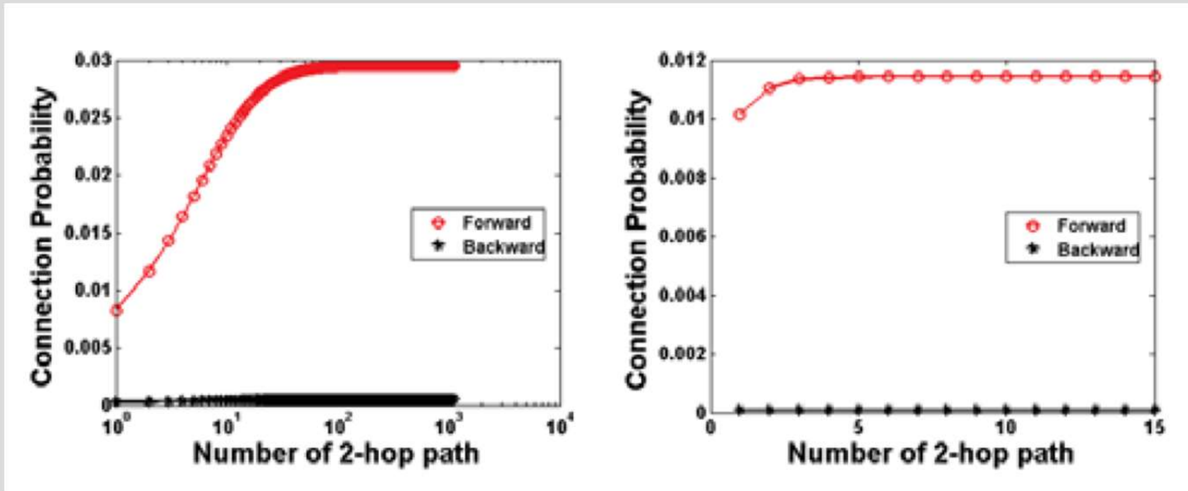
Nodes → Vectors

Advantages:
- Apply classic vector-based machine learning techniques to process graph data;
- Facilitate parallelization of graph analysis;

Application:
- Graph reconstruction
- Link prediction
- Classification …

Asymmetric transitivity in Social Network





INTUITION & SOLUTION

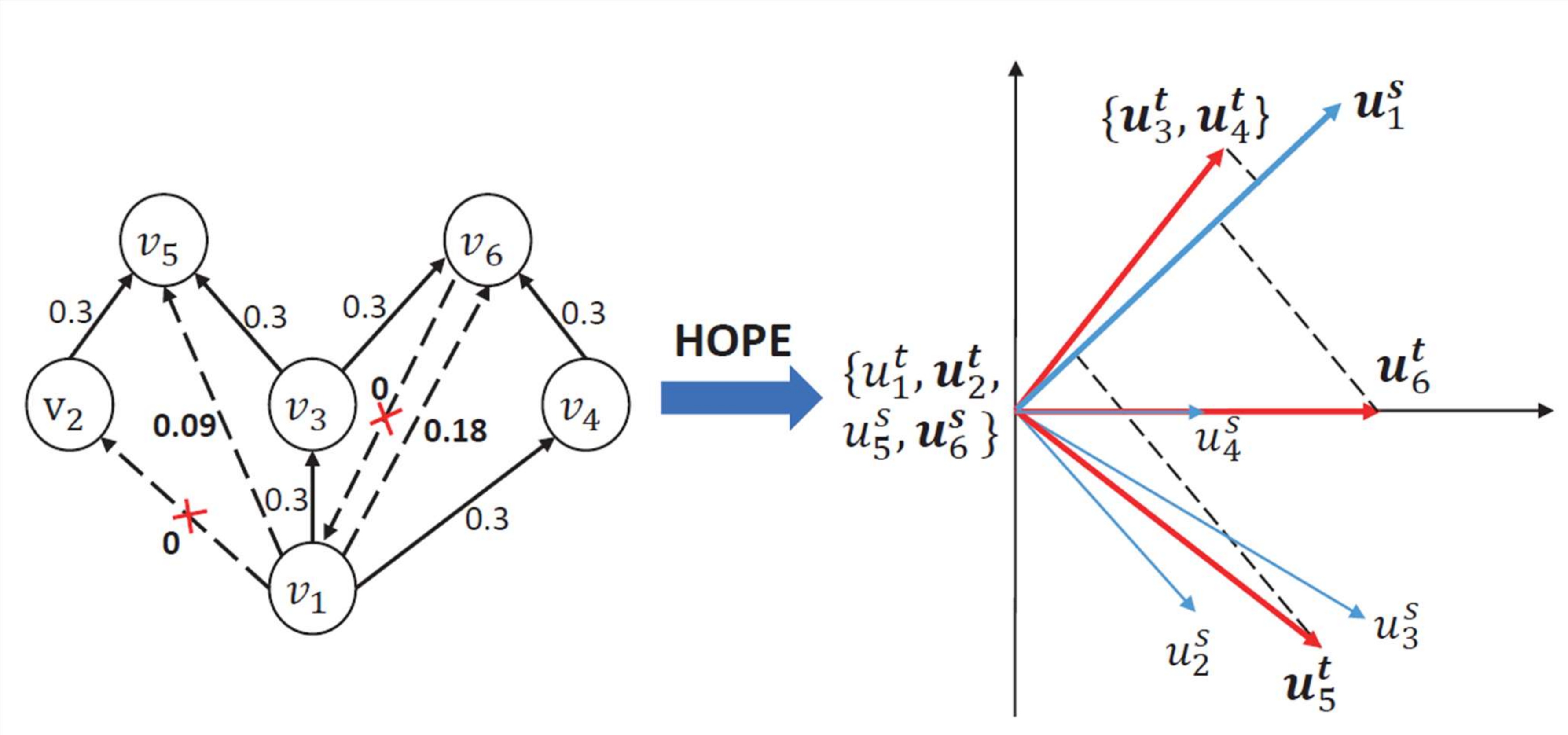
Node → Source vector / Target vector

The more and the shorter paths from v_i to v_j,
the more similar should be v_i's source vector and v_j's target vector.

Embedding Vectors $\xrightarrow{\text{Approximate}}$ High-Order Proximity Matrix

High-Order Proximity

EXAMPLE



v_5
v_6
0.3
0.3
0.3
0.3
v_2
v_3
v_4
0.09
0
0.18
0.3
0
0.3
v_1
HOPE
$\{\boldsymbol{u}_3^t, \boldsymbol{u}_4^t\}$
$\boldsymbol{u}_1^s$
$\{u_1^t, \boldsymbol{u}_2^t, u_5^s, \boldsymbol{u}_6^s\}$
$\boldsymbol{u}_6^t$
u_4^s
u_2^s
$\boldsymbol{u}_5^t$
u_3^s

LINEAR ALGEBRA BASIS

Matrix Inverse: $AA^{-1} = A^{-1}A = I$

Nonsingular matrix: one that has a matrix inverse.

Matrix transpose: $(ABC)^T = C^T B^T A^T$

Diagonal matrix: Any square diagonal matrix is a symmetric matrix: $A^T = A$

Orthogonal matrix: $O^T = O^{-1}$

Eigenvalues and eigenvectors: $A\vec{e}_\lambda = \lambda\vec{e}_\lambda$

Eigendecomposition: $A = Q\Lambda Q^{-1}$, for normal matrices $A = O\Lambda O^{-1}$

Singular Value Decomposition (SVD): $A = U\Sigma V^T$

HIGH-ORDER PROXIMITY PRESERVED EMBEDDING

Notations:

$G = \{V, E\}$

$V = \{v_1, \ldots v_i, \ldots v_N\}$, N is the number of vertexes.

$e_{ij} = (v_i, v_j) \in E$ represents a directed edge form v_i to v_j

A: adjacency matrix (NxN)

S: high-order proximity matrix (NxN), S_{ij} is the proximity between v_i and v_j

$U = [U^s, U^t]$ is the embedding matrix, u_i is the embedding vector for v_i.

$U^s, U^t \in R^{NxK}$ are the source embedding vectors and target embedding vectors, K is embedding dimensions.

HIGH-ORDER PROXIMITY PRESERVED EMBEDDING

Problem Definition:

Embedding to approximate high-order proximity

$$\min \|\mathbf{S} - \mathbf{U}^s \cdot \mathbf{U}^{t^\top}\|_F^2 \quad (1)$$

HIGH-ORDER PROXIMITY PRESERVED EMBEDDING

High-order proximity (S):

- Katz Index
- Rooted PageRank
- Common Neighbours
- Adamic-Adar

$$\mathbf{S}^{Katz} = \sum_{l=1}^{\infty} (\beta \cdot \mathbf{A})^l = \beta \cdot \mathbf{A} \cdot \mathbf{S}^{Katz} + \beta \cdot \mathbf{A} \quad (3)$$

$$\mathbf{S}^{RPR} = \alpha \cdot \mathbf{S}^{RPR}_{ij} \cdot \mathbf{P} + (1 - \alpha) \cdot \mathbf{I} \quad (7)$$

$$\mathbf{S}^{CN} = \mathbf{A}^2 \quad (11)$$

$$\mathbf{S}^{AA} = \mathbf{A} \cdot \mathbf{D} \cdot \mathbf{A} \quad (14)$$

$$\mathbf{S} = \mathbf{M}_g^{-1} \cdot \mathbf{M}_l \quad (2)$$

Table 1: General Formulation for High-order Proximity Measurements

	Proximity Measurement	$\mathbf{M}_g$	$\mathbf{M}_l$
Global	Katz	$\mathbf{I} - \beta \cdot \mathbf{A}$	$\beta \cdot \mathbf{A}$
	Personalized Pagerank	$\mathbf{I} - \alpha\mathbf{P}$	$(1 - \alpha) \cdot \mathbf{I}$
Local	Common neighbors	$\mathbf{I}$	$\mathbf{A}^2$
	Adamic-Adar	$\mathbf{I}$	$\mathbf{A} \cdot \mathbf{D} \cdot \mathbf{A}$

HIGH-ORDER PROXIMITY PRESERVED EMBEDDING

$$\min \|\mathbf{S} - \mathbf{U}^s \cdot \mathbf{U}^{t\top}\|_F^2 \tag{1}$$

The objective is to find an optimal rank-K approximation of S

Solution: Performing SVD on S, and using the largest K singular value and corresponding singular vectors to construct the optimal embedding vectors.

$$\mathbf{S} = \sum_{i=1}^{N} \sigma_i \mathbf{v}_i^s \mathbf{v}_i^{t\top} \tag{18}$$

Thus, the vectors we aim to find are:

Too expensive!

$$\mathbf{U}^s = [\sqrt{\sigma_1} \cdot \mathbf{v}_1^s, \cdots, \sqrt{\sigma_K} \cdot \mathbf{v}_K^s] \tag{19}$$

$$\mathbf{U}^t = [\sqrt{\sigma_1} \cdot \mathbf{v}_1^t, \cdots, \sqrt{\sigma_K} \cdot \mathbf{v}_K^t] \tag{20}$$

HIGH-ORDER PROXIMITY PRESERVED EMBEDDING

THEOREM 1. *If we have the singular value decomposition of the general formulation*

$$\mathbf{M}_g^{-1} \cdot \mathbf{M}_l = \mathbf{V}^s \Sigma \mathbf{V}^{t\top}$$

, where $\mathbf{V}^t$ and $\mathbf{V}^s$ are two orthogonal matrices,

$$\Sigma = diag(\sigma_1, \sigma_2, \cdots, \sigma_N)$$

.

Then, there exists a nonsingular matrix $\mathbf{X}$ and two diagonal matrices, i.e. Σ^l and Σ^g, satisfying that

$$\boxed{\begin{aligned} \mathbf{V}^{t\top} \mathbf{M}_l^\top \mathbf{X} &= \Sigma^l \\ \mathbf{V}^{s\top} \mathbf{M}_g^\top \mathbf{X} &= \Sigma^g \end{aligned}}$$

, where

$$\begin{gathered} \Sigma^l = diag(\sigma_1^l, \sigma_2^l, \cdots, \sigma_N^l) \\ \Sigma^g = diag(\sigma_1^g, \sigma_2^g, \cdots, \sigma_N^g) \\ \sigma_1^l \geq \sigma_2^l \geq \cdots \geq \sigma_K^l \geq 0 \\ 0 \leq \sigma_1^g \leq \sigma_2^g \leq \cdots \leq \sigma_K^g \end{gathered}$$

$$\boxed{\sigma_i = \frac{\sigma_i^l}{\sigma_i^g}} \tag{21}$$

HIGH-ORDER PROXIMITY PRESERVED EMBEDDING

Instead of performing standard SVD on S, to get

$\{\mathbf{v}_1^s, \cdots, \mathbf{v}_K^s\}$ $\{\mathbf{v}_1^t, \cdots, \mathbf{v}_K^t\}$ $\{\sigma_1, \cdots, \sigma_K\}$

Recall, $\mathbf{S} = \mathbf{M}_g^{-1} \cdot \mathbf{M}_l$ and through performing JDGSVD on Mg and Ml, we can get:

$\{\sigma_1^g, \cdots, \sigma_K^g\}$ $\{\sigma_1^l, \cdots, \sigma_K^l\}$ $\{\mathbf{v}_1^s, \cdots, \mathbf{v}_K^s\}$ $\{\mathbf{v}_1^t, \cdots, \mathbf{v}_K^t\}$

And calculating $\{\sigma_1, \cdots, \sigma_K\}$ through

$$\sigma_i = \frac{\sigma_i^l}{\sigma_i^g} \tag{21}$$

HIGH-ORDER PROXIMITY PRESERVED EMBEDDING

Algorithm 1 High-order Proximity preserved Embedding

Require: adjacency matrix $\mathbf{A}$, embedding dimension K, parameters of high-order proximity measurementθ.

Ensure: embedding source vectors $\mathbf{U}^s$ and target vectors $\mathbf{U}^t$.

1: calculate $\mathbf{M}_g$ and $\mathbf{M}_l$.

2: perform JDGSVD with $\mathbf{M}_g$ and $\mathbf{M}_l$, and obtain the generalized singular values $\{\sigma_1^l, \cdots, \sigma_K^l\}$ and $\{\sigma_1^g, \cdots, \sigma_K^g\}$, and the corresponding singular vectors, $\{\mathbf{v}_1^s, \cdots, \mathbf{v}_K^s\}$ and $\{\mathbf{v}_1^t, \cdots, \mathbf{v}_K^t\}$.

3: calculate singular values $\{\sigma_1, \cdots, \sigma_K\}$ according to Equation (21).

4: calculate embedding matrices $\mathbf{U}^s$ and $\mathbf{U}^t$ according to Equation (19) and (20).

HIGH-ORDER PROXIMITY PRESERVED EMBEDDING

Approximation Error

THEOREM 2. *Given the proximity matrix,* $\mathbf{S}$*, of a directed graph, and the embedding vectors,* $\mathbf{U}^s$ *and* $\mathbf{U}^t$*, learned by* HOPPE. *Then the approximation error is*

$$\|\mathbf{S} - \mathbf{U}^s \cdot \mathbf{U}^{t^T}\|_F^2 = \sqrt{\sum_{i=K+1}^{N} \sigma_i^2}$$

, and the relative approximation error is:

$$\frac{\|\mathbf{S} - \mathbf{U}^s \cdot \mathbf{U}^{t^T}\|_F^2}{\|\mathbf{S}\|_F^2} = \sqrt{\frac{\sum_{i=K+1}^{N} \sigma_i^2}{\sum_{i=1}^{N} \sigma_i^2}} \qquad (22)$$

where $\{\sigma_i\}$ *are the singular values of* $\mathbf{S}$ *in descend order.*

EXPERIMENT

1. Dataset:

	Syn	Cora	SN-Twitter	SN-TWeibo
|V|	10,000	23166	465,017	1,944,589
|E|	144,555	91500	834,797	50,655,143

2. Baseline methods:

LINE, DeepWalk, PPE, Common Neighbors, Adamic-Adar

3. Applications:

- Error of approximation — $RMSE = \sqrt{\frac{\|\mathbf{S} - \mathbf{U}^s {\mathbf{U}^t}^\top\|_F^2}{N^2}}$
- Graph reconstruction
- Link prediction

$$Precision@k = \frac{|\{(i,j)|(i,j) \in \mathbf{E}_p \cap \mathbf{E}_o\}|}{|\mathbf{E}_p|}$$

- Vertex recommendation — $AP@k(i) = \frac{\sum_{j=1}^{k} P_i(j) \cdot \delta_i(j)}{\sum_{j=1}^{k} \delta_i(j)}$

EXPERIMENT

Error of proximity approximation

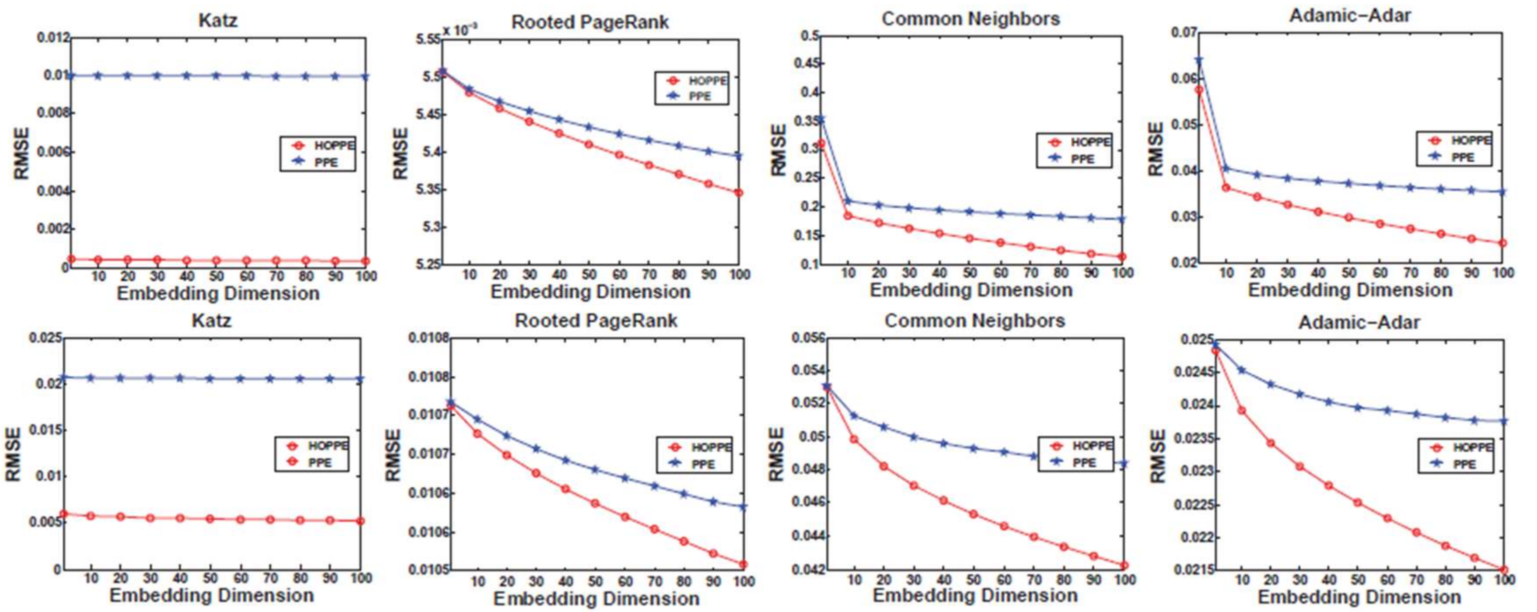


Figure 3: Error of proximity approximation. We evaluate the errors of HOPE and PPE in approximating four proximity measurements, including Katz, RPR, Common Neighbors and Adamic-Adar. First row is the results on Synthetic Data, and second row is the results on Cora. For Katz, $\beta = 0.1$; for RPR, $\alpha = 0.5$.

EXPERIMENT

Graph Reconstruction

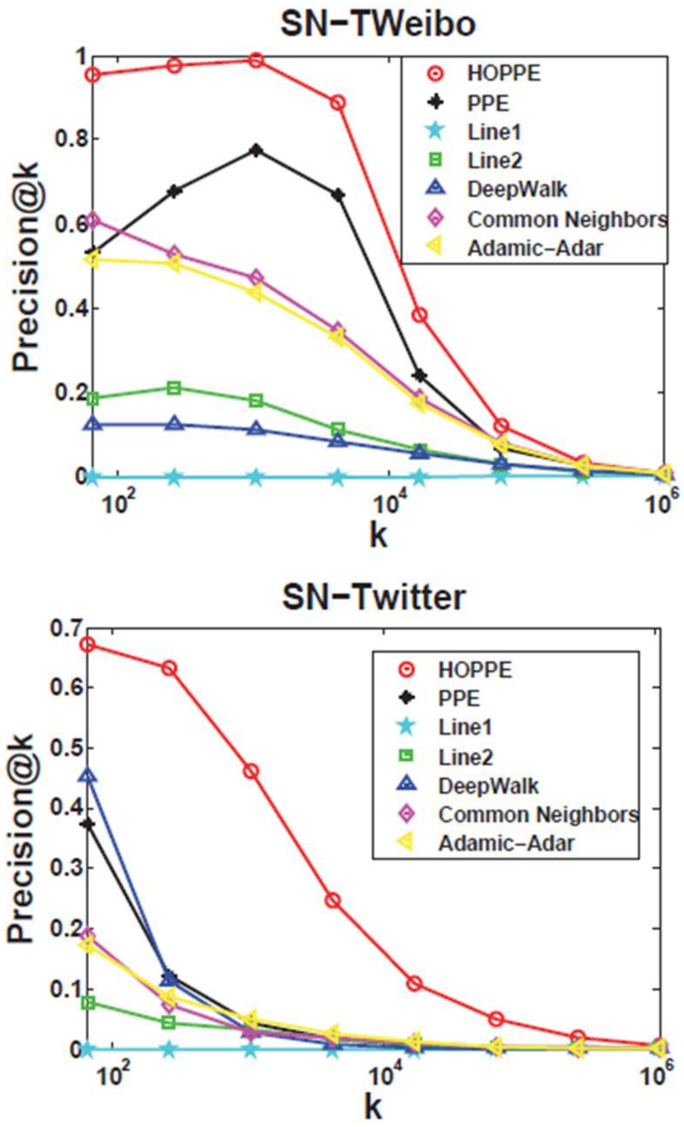


Figure 5: Precision@k of graph reconstruction on SN-TWeibo and SN-Twitter. We rank pairs of ver-

EXPERIMENT

Link Prediction

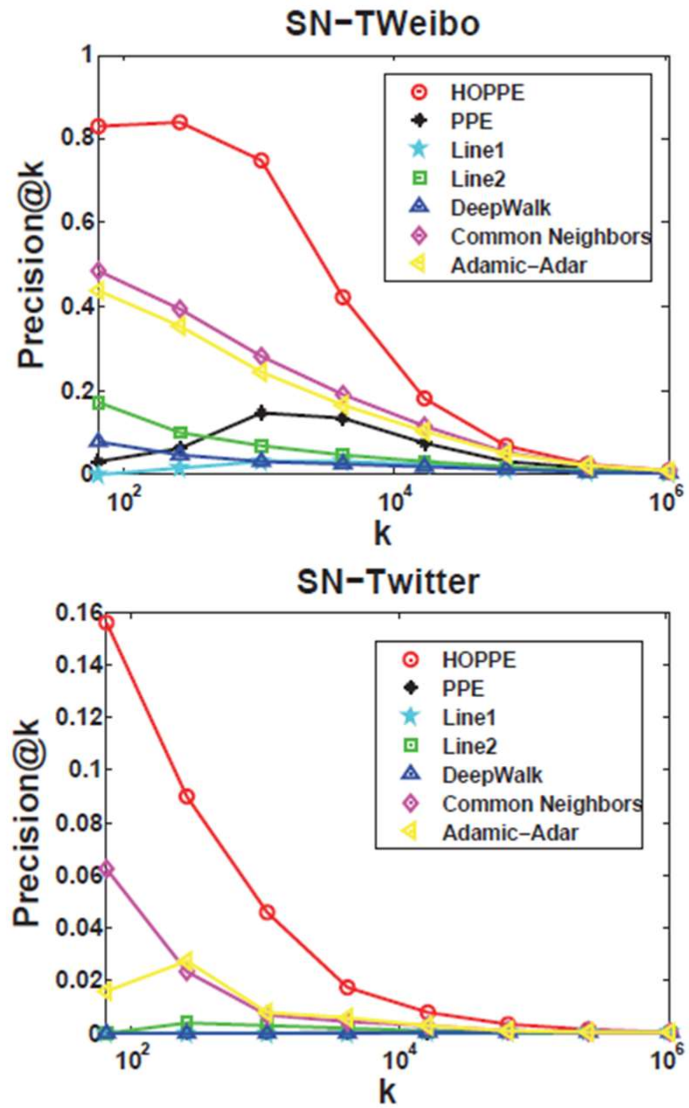


Figure 6: Precision@k of link prediction on SN-TWeibo and SN-Twitter. We rank pairs of vertexes

EXPERIMENT

Vertex Recommendation

Table 3: MAP of vertex recommendation on SN-TWeibo and SN-Twitter. For each vertex, the recommended vertex list is ranked according to the predicted proximity between vertexes. For embedding algorithms, we calculate the predicted proximity by performing inner product between embedding vectors.

Method	SN-TWebio			SN-Twitter		
	MAP@10	MAP@50	MAP@100	MAP@10	MAP@50	MAP@100
HOPE	**0.2295**	**0.1869**	**0.169**	**0.1000**	**0.0881**	**0.0766**
PPE	0.0928	0.0845	0.077	0.0061	0.0077	0.0081
LINE1	0	0	0.005	0.0209	0.0221	0.0221
LINE2	0.051	0.051	0.048	0.0044	0.0043	0.0035
DeepWalk	0.0635	0.0583	0.004	0.0006	0.0008	0.001
Common Neighbors	0.1217	0.1031	0.155	0.0394	0.0379	0.0369
Adamic-Adar	0.1173	0.0990	0.156	0.0455	0.0442	0.0423

CONCLUSION

Contributions:

Embedding Vectors $\xrightarrow{\text{Approximate}}$ High-Order Proximity Matrix

HOPE Algorithm