Geometric Graph Representation Learning via Maximizing Rate Reduction

Xiaotian Han[1], Zhimeng Jiang[1], Ninghao Liu[2], Qingquan Song[3], Jundong Li[4], Xia Hu[5]

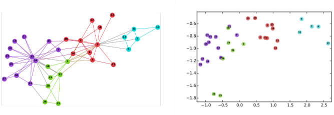
[1]Texas A&M University, [2]University of Georgia, [3]LinkedIn, [4]University of Virginia, [5]Rice University

April 27, 2022

Overview

1. Motivation
2. Preliminaries
   - Coding Rate
3. Methodology
   - Rate Reduction for Graphs
   - Objective Function of $\mathrm{G}^2\mathrm{R}$
   - Theoretical Justification
4. Experiments
   - Verification Experiments
   - Performance Experiments

1. Graph Representation Learning is to map no-Euclidean graph data to low-dimensional vector.



2. Methods
   1. Random Walk-based: DeepWalk(Perozzi et al., 2014), Node2vec(Grover & Leskovec, 2016)

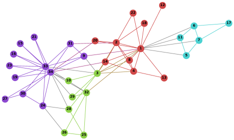
$$Pr(n_i|f(u)) = \frac{e^{f(n_i)\cdot f(u)}}{\sum_{v\in V} e^{f(v)\cdot f(u)}}.$$

   2. Contrastive Learning: GRACE(Zhu et al., 2020), GraphCL(You et al., 2020)

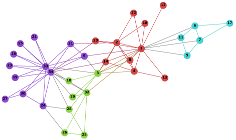
$$\ell(\boldsymbol{u}_i, \boldsymbol{v}_i) = \log \frac{e^{\theta(\boldsymbol{u}_i,\boldsymbol{v}_i)/\tau}}{e^{\theta(\boldsymbol{u}_i,\boldsymbol{v}_i)/\tau} + \sum_{k\in N} 1_{[k\neq i]} e^{\theta(\boldsymbol{u}_i,\boldsymbol{v}_k)/\tau}}.$$

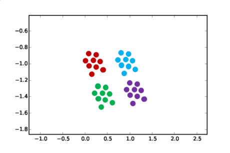
3. Key Idea: model the similarity of connected nodes.

Motivation



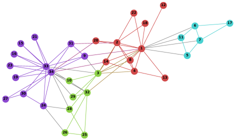
Motivation

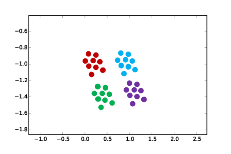




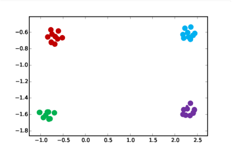
Less diversity of node representations as a whole

Motivation



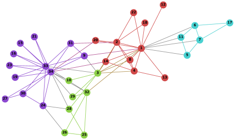


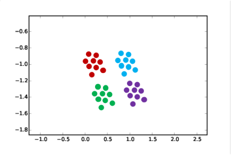
Less diversity of node representations as a whole



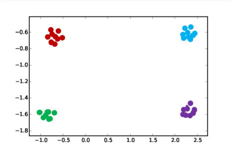
Less diversity of node representation within groups

Motivation





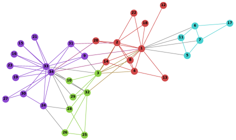
Less diversity of node representations as a whole

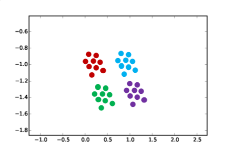


Less diversity of node representation within groups

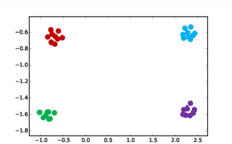
- We argue that previous methods encourage the *local* similarity between connected nodes, but could fail to capture the *global* distribution of node representations.

Motivation



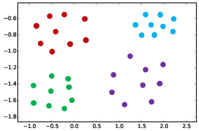


Less diversity of node representations as a whole



Less diversity of node representation within groups

- We argue that previous methods encourage the *local* similarity between connected nodes, but could fail to capture the *global* distribution of node representations.



The ideal node representation should be in geometrically-good representation space.

Geometric Graph Representation Learning

In this paper, we propose Geometric Graph Representation Learning ($\mathrm{G^2R}$).
We propose the following desirable properties for node representations

1. The whole node representation should be diverse.
2. The node representation within groups should be similar but span their own subspaces.

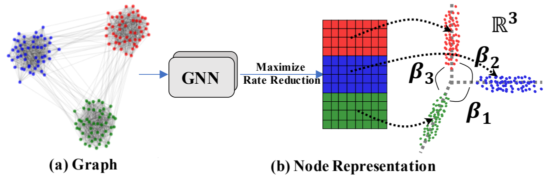


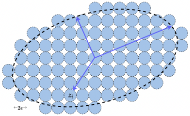
Figure: Overview of $\mathrm{G^2R}$. It maps nodes in distinct groups into different subspaces, while each subspace is compact and different subspaces are dispersedly distributed.

Coding Rate[2] (Ma et al., 2007)

Suppose we have a bunch of data representations $\mathbf{W} = (w_1, w_2, \cdots, w_m)$, Then the number of bit needed to encode the data $\mathbf{W}$ is [1]

$$R(\mathbf{W}) \doteq \frac{1}{2}\log_2 \det(\mathbf{I} + \frac{n}{m\epsilon^2}\mathbf{W}\mathbf{W}^\top). \tag{1}$$

$R(\mathbf{W})$ is an intrinsic measure for the volume of $\mathbf{W}$.



$\mathrm{vol}(Z)$

[1] ϵ is the error allowable for encoding every vector w_i in $\mathbf{W}$.

[2] This slide is largely based on Yi Ma slides at https://book-wright-ma.github.io/Lecture-Slides/Lecture_21_22.pdf

Coding Rate of Graphs

First, the whole node representation should be diverse.
Considering the graph neural network to map the graph $\mathcal{G} = (\mathbf{A}, \mathbf{X})$ to node representation $\mathbf{Z}$,

$$\mathbf{A} \in \mathbb{R}^{N \times N}, \mathbf{X} \in \mathbb{R}^{D \times N} \xrightarrow{\mathsf{GNN}(\mathbf{A}, \mathbf{X} | \theta)} \mathbf{Z} \in \mathbb{R}^{d \times N}.$$

Coding rate can be applied to estimate the number of bits for representing $\mathbf{Z}$,

$$R_{\mathcal{G}}(\mathbf{Z}, \epsilon) \doteq \frac{1}{2} \log \det \left(\mathbf{I} + \frac{d}{N \epsilon^2} \mathbf{Z} \mathbf{Z}^\top \right), \quad (2)$$

Coding Rate of Graphs

First, the whole node representation should be diverse.
Considering the graph neural network to map the graph $\mathcal{G} = (\mathbf{A}, \mathbf{X})$ to node representation $\mathbf{Z}$,

$$\mathbf{A} \in \mathbb{R}^{N \times N}, \mathbf{X} \in \mathbb{R}^{D \times N} \xrightarrow{\text{GNN}(\mathbf{A}, \mathbf{X} | \theta)} \mathbf{Z} \in \mathbb{R}^{d \times N}.$$

Coding rate can be applied to estimate the number of bits for representing $\mathbf{Z}$,

$$R_{\mathcal{G}}(\mathbf{Z}, \epsilon) \doteq \frac{1}{2} \log \det \left(\mathbf{I} + \frac{d}{N \epsilon^2} \mathbf{Z} \mathbf{Z}^{\top} \right), \tag{2}$$

larger $R_{\mathcal{G}}$ $\rightarrow$ more bits in representations $\rightarrow$ diverse representations.

Coding Rate for Node Groups

Second, the node representation within groups should be similar but span their own subspaces.

For one node i, we compute the average coding rate for the neighbor of node i (as a group) as follows:

$$R_{\mathcal{G}}^{c}(\mathbf{Z}, \epsilon | \mathbf{A}_i) \doteq \frac{\operatorname{tr}(\mathbf{A}_i)}{2N} \cdot \log\det\left(\mathbf{I} + \frac{d}{\operatorname{tr}(\mathbf{A}_i)\epsilon^2}\mathbf{Z}\mathbf{A}_i\mathbf{Z}^{\top}\right). \tag{3}$$

Where $\mathbf{A}_i$ indicate the the neighbors of node i (as a group).

For all nodes, the average of the coding rate is as following:

$$R_{\mathcal{G}}^{c}(\mathbf{Z}, \epsilon | \mathcal{A}) \doteq \frac{1}{\bar{d}} \sum_{i=1}^{N} R_{\mathcal{G}}^{c}(\mathbf{Z}, \epsilon | \mathbf{A}_i). \tag{4}$$

Coding Rate for Node Groups

Second, the node representation within groups should be similar but span their own subspaces.

For one node i, we compute the average coding rate for the neighbor of node i (as a group) as follows:

$$R^c_{\mathcal{G}}(\mathbf{Z}, \epsilon | \mathbf{A}_i) \doteq \frac{\operatorname{tr}(\mathbf{A}_i)}{2N} \cdot \log\det\left(\mathbf{I} + \frac{d}{\operatorname{tr}(\mathbf{A}_i)\epsilon^2}\mathbf{Z}\mathbf{A}_i\mathbf{Z}^\top\right). \tag{3}$$

Where $\mathbf{A}_i$ indicate the the neighbors of node i (as a group).

For all nodes, the average of the coding rate is as following:

$$R^c_{\mathcal{G}}(\mathbf{Z}, \epsilon | \mathcal{A}) \doteq \frac{1}{\bar{d}} \sum_{i=1}^{N} R^c_{\mathcal{G}}(\mathbf{Z}, \epsilon | \mathbf{A}_i). \tag{4}$$

smaller $R^c_{\mathcal{G}}$ $\rightarrow$ less bits in representations $\rightarrow$ similar representations.

Objective Function

To enforce

1. diverse node representations space
2. more similar representations for connected nodes

We propose to maximize the following objective function

$$\begin{aligned}
&\Delta R_{\mathcal{G}}(\mathbf{Z}, \mathbf{A}, \epsilon) \\
&= R_{\mathcal{G}}(\mathbf{Z}, \epsilon) - R^c_{\mathcal{G}}(\mathbf{Z}, \epsilon \mid \mathcal{A}) \\
&\doteq \frac{1}{2} \log\det\left(\mathbf{I} + \frac{d}{N\epsilon^2}\mathbf{Z}\mathbf{Z}^\top\right) - \frac{1}{\bar{d}} \sum_{i=1}^{N} \frac{\operatorname{tr}(\mathbf{A}_i)}{2N} \cdot \log\det\left(\mathbf{I} + \frac{d}{\operatorname{tr}(\mathbf{A}_i)\epsilon^2}\mathbf{Z}\mathbf{A}_i\mathbf{Z}^\top\right)
\end{aligned} \tag{5}$$

Since $\mathbf{Z} = \text{GNN}(\mathbf{X}, \mathbf{A}|\theta)$ and the parameters θ will be optimized by

$$\max_{\theta} \Delta R_{\mathcal{G}}(\mathbf{X}, \mathbf{A}, \epsilon),$$

What is $\mathrm{G}^2\mathrm{R}$ Doing Theoretically?

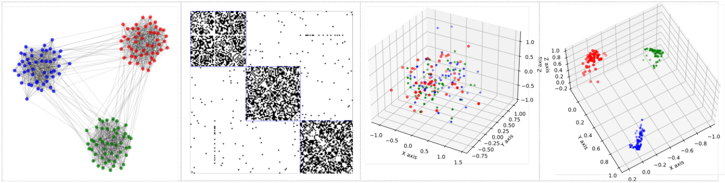
Considering the principal angle between subspaces and decomposition the adjacency matrix, the rate reduction will take

$$\Delta R_{\mathcal{G}} = \sum_{j=1}^{2} \log \left(\frac{\det^{\frac{1}{4}} \left(\mathbf{I} + \frac{d}{N\epsilon^2} \mathbf{Z}_j^\top \mathbf{Z}_j \right)}{\det^{\frac{p^i - p^o}{2N}} \left(\mathbf{I} + \frac{d}{M\epsilon^2} \mathbf{Z}_j^\top \mathbf{Z}_j \right)} \right) + \frac{1}{2} \cdot \log \beta. \tag{6}$$

Maximizing the second term (principal angle β of different subspaces) will:

- **Inter-communities.** The node representations of different communities lie in different subspaces and the principal angle of them are maximized (i.e., nearly pairwise orthogonal).
- **Intra-communities.** The representations of nodes in the same community should be more similar than nodes in different communities (in the same subspace).

Experiments with Synthetic Data



(a) Synthetic graph **(b) Adjacency matric** **(c) Original node features** **(d) Learned node representation**

Figure: Synthetic graph and visualization of its node features and representations. The different colors in (a)(c)(d) indicate different communities. The learned node representations in (d) are 3-dimensional vectors obtained by $\mathrm{G^2R}$.

We make the following observations:

1. The learned node representations in different communities are nearly orthogonal in the three-dimensional space.
2. The node representations in the same community are compact.

Will Representation Learned by $\mathrm{G^2R}$ (nearly) Orthogonal? Visualization Analysis

We perform a visualization experiment to analyze the representations learned by $\mathrm{G^2R}$ to verify the orthogonality of the learned representation. We plot two classes of nodes in each figure.

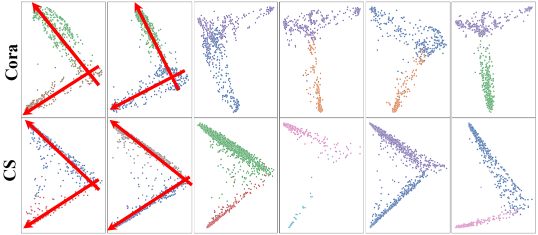


Figure: PCA visualization of learned representations.

We make the following observations:

1. The representations of nodes in different classes learned by $\mathrm{G^2R}$ are nearly orthogonal to each other.

Performance Comparison to Unsupervised Methods

Table: Performance comparison to unsupervised methods. The best performance among baselines is underlined. The best performance is in **boldface**.

Statistic		Cora		CiteSeer		PubMed		CoraFull	CS	Physics	Computers	Photo
Metric	Feature	Public	Random	Public	Random	Public	Random	Random	Random	Random	Random	Random
Feature	$\mathbf{X}$	58.90	60.19	58.69	61.70	69.96	73.90	40.06	88.14	87.49	67.48	59.52
PCA	$\mathbf{X}$	57.91	59.90	58.31	60.00	69.74	74.00	38.46	88.59	87.66	72.65	57.45
SVD	$\mathbf{X}$	58.57	60.21	58.10	60.80	69.89	73.79	38.64	88.55	87.98	68.17	60.98
isomap	$\mathbf{X}$	40.19	44.60	18.20	18.90	62.41	63.90	4.21	73.68	82.84	72.66	44.00
LLE	$\mathbf{X}$	29.34	36.70	18.26	21.80	52.82	54.00	5.70	72.23	81.35	45.29	35.37
DeepWalk	$\mathbf{A}$	74.03	73.76	48.04	51.80	68.72	71.28	51.65	83.25	88.08	86.47	76.58
Node2vec	$\mathbf{A}$	73.64	72.54	46.95	49.37	70.17	68.70	50.35	82.12	86.77	85.15	75.67
DeepWalk+F	$\mathbf{X}, \mathbf{A}$	77.36	77.62	64.30	66.96	69.65	71.84	54.63	83.34	88.15	**86.49**	65.97
Node2vec+F	$\mathbf{X}, \mathbf{A}$	75.44	76.84	63.22	66.75	70.6	69.12	54.00	82.20	86.86	85.15	65.01
GAE	$\mathbf{X}, \mathbf{A}$	73.68	74.30	58.21	59.69	76.16	80.08	42.54	88.88	91.01	37.72	48.72
VGAE	$\mathbf{X}, \mathbf{A}$	77.44	76.42	59.53	60.37	78.00	77.75	53.69	88.66	90.33	49.09	48.33
DGI	$\mathbf{X}, \mathbf{A}$	81.26	82.11	69.50	70.15	77.70	79.06	53.89	91.22	92.12	79.62	70.65
GRACE	$\mathbf{X}, \mathbf{A}$	80.46	80.36	68.72	68.04	80.67	OOM	53.95	90.04	OOM	81.94	70.38
GraphCL	$\mathbf{X}, \mathbf{A}$	81.89	81.12	68.40	69.67	OOM	81.41	OOM	OOM	OOM	79.90	OOM
GMI	$\mathbf{X}, \mathbf{A}$	80.28	81.20	65.99	70.50	OOM	OOM	OOM	OOM	OOM	52.36	OOM
$\mathrm{G^2R}$**(ours)**	$\mathbf{X}, \mathbf{A}$	**82.58**	**83.32**	**71.2**	**70.66**	**81.69**	**81.69**	**59.70**	**92.64**	**94.93**	82.24	**90.68**

We make the following observations:

1. $\mathrm{G^2R}$ outperforms all baselines by significant margins on seven datasets.

G^2R is even Better than Supervised Counterparts

Table: Comparison to supervised baselines on node classification. The ‘Avg.Rank’ is the average rank among all the methods on all datasets.

Methods	Cora		CiteSeer		PubMed		CS	Physics	Computers	Photo	Avg.Rank
	Public	Random	Public	Random	Public	Random					
LogReg	52.0	58.3	55.8	60.8	73.6	69.7	86.4	86.7	64.1	73.0	**11.3**
MLP	61.6	59.8	61.0	58.8	74.2	70.1	88.3	88.9	44.9	69.6	**10.9**
LP	71.0	79.0	50.8	65.8	70.6	73.3	73.6	86.6	70.8	72.6	**11.2**
LP NL	71.2	79.7	51.2	66.9	72.6	77.8	76.7	86.8	75.0	83.9	**9.5**
ChebNet	80.5	76.8	69.6	67.5	78.1	75.3	89.1	-	15.2	25.2	**10.0**
GCN	81.3	79.1	71.1	68.2	78.8	77.1	91.1	92.8	82.6	91.2	**5.7**
GAT	83.1	80.8	70.8	68.9	79.1	77.8	90.5	92.5	78.0	85.7	**5.8**
MoNet	79.0	84.4	70.3	71.4	78.9	83.3	90.8	92.5	83.5	91.2	**4.0**
SAGE	78.0	84.0	70.1	71.1	78.8	79.2	91.3	93.0	82.4	91.4	**4.7**
APPNP	83.3	81.9	71.8	69.8	80.1	79.5	90.1	90.9	20.6	30.0	**6.0**
SGC	81.7	80.4	71.3	68.7	78.9	76.8	90.8	-	79.9	90.7	**5.9**
DAGNN	84.4	83.7	73.3	71.2	80.5	80.1	92.8	94.0	84.5	92.0	**1.7**
Ours	83.3	82.6	70.6	71.2	81.7	81.7	92.6	94.9	82.2	90.7	**3.1**

We make the following observations:

1. G^2R shows comparable performance across all seven datasets, although the baselines are all supervised methods.

References I

Grover, A., & Leskovec, J. (2016). Node2vec: Scalable feature learning for networks. *Proceedings of the 22nd ACM SIGKDD international conference on Knowledge discovery and data mining*, 855–864.

Ma, Y., Derksen, H., Hong, W., & Wright, J. (2007). Segmentation of multivariate mixed data via lossy data coding and compression. *IEEE transactions on pattern analysis and machine intelligence*, *29*(9), 1546–1562.

Perozzi, B., Al-Rfou, R., & Skiena, S. (2014). Deepwalk: Online learning of social representations. *Proceedings of the 20th ACM SIGKDD international conference on Knowledge discovery and data mining*, 701–710.

You, Y., Chen, T., Sui, Y., Chen, T., Wang, Z., & Shen, Y. (2020). Graph contrastive learning with augmentations. *NeurIPS*.

Zhu, Y., Xu, Y., Yu, F., Liu, Q., Wu, S., & Wang, L. (2020). Deep graph contrastive representation learning. *arXiv preprint arXiv:2006.04131*.

Geometric Graph Representation Learning via Maximizing Rate Reduction

Thanks all for your attendance!!

For more details, please check out our paper at
https://doi.org/10.1145/3485447.3512170