Simplifying Graph Convolutional Networks (Supplementary Material)

Felix Wu [*1] **Tianyi Zhang** [*1] **Amauri Holanda de Souza Jr.** [*1 2] **Christopher Fifty** [1] **Tao Yu** [1]
Kilian Q. Weinberger [1]

A. The spectrum of $\tilde{\mathbf{\Delta}}_{\text{sym}}$

The normalized Laplacian defined on graphs with self-loops, $\tilde{\mathbf{\Delta}}_{\text{sym}}$, consists of an instance of generalized graph Laplacians and hold the interpretation as a difference operator, i.e. for any signal $\mathbf{x} \in \mathbb{R}^n$ it satisfies

$$(\tilde{\mathbf{\Delta}}_{\text{sym}}\mathbf{x})_i = \sum_j \frac{\tilde{a}_{ij}}{\sqrt{d_i+\gamma}} \left(\frac{x_i}{\sqrt{d_i+\gamma}} - \frac{x_j}{\sqrt{d_j+\gamma}} \right).$$

Here, we prove several properties regarding its spectrum.

Lemma 1. *(Non-negativity of $\tilde{\mathbf{\Delta}}_{sym}$) The augmented normalized Laplacian matrix is symmetric positive semi-definite.*

Proof. The quadratic form associated with $\tilde{\mathbf{\Delta}}_{\text{sym}}$ is

$$\begin{aligned}
\mathbf{x}^\top \tilde{\mathbf{\Delta}}_{\text{sym}}\mathbf{x} &= \sum_i x_i^2 - \sum_i \sum_j \frac{\tilde{a}_{ij}x_ix_j}{\sqrt{(d_i+\gamma)(d_j+\gamma)}} \\
&= \frac{1}{2}\left(\sum_i x_i^2 + \sum_j x_j^2 - \sum_i \sum_j \frac{2\tilde{a}_{ij}x_ix_j}{\sqrt{(d_i+\gamma)(d_j+\gamma)}} \right) \\
&= \frac{1}{2}\left(\sum_i \sum_j \frac{\tilde{a}_{ij}x_i^2}{d_i+\gamma} + \sum_j \sum_i \frac{\tilde{a}_{ij}x_j^2}{d_j+\gamma} \right. \\
&\quad \left. - \sum_i \sum_j \frac{2\tilde{a}_{ij}x_ix_j}{\sqrt{(d_i+\gamma)(d_j+\gamma)}} \right) \\
&= \frac{1}{2}\sum_i \sum_j \tilde{a}_{ij} \left(\frac{x_i}{\sqrt{d_i+\gamma}} - \frac{x_j}{\sqrt{d_j+\gamma}} \right)^2 \geq 0 \qquad (1)
\end{aligned}$$

□

Lemma 2. 0 *is an eigenvalue of both $\mathbf{\Delta}_{sym}$ and $\tilde{\mathbf{\Delta}}_{sym}$.*

Proof. First, note that $\mathbf{v} = [1, \dots, 1]^\top$ is an eigenvector of $\mathbf{\Delta}$ associated with eigenvalue 0, i.e., $\mathbf{\Delta v} = (\mathbf{D} - \mathbf{A})\mathbf{v} = \mathbf{0}$.

Also, we have that $\tilde{\mathbf{\Delta}}_{\text{sym}} = \tilde{\mathbf{D}}^{-1/2}(\tilde{\mathbf{D}} - \tilde{\mathbf{A}})\tilde{\mathbf{D}}^{-1/2} = \tilde{\mathbf{D}}^{-1/2}\mathbf{\Delta}\tilde{\mathbf{D}}^{-1/2}$. Denote $\mathbf{v}_1 = \tilde{\mathbf{D}}^{1/2}\mathbf{v}$, then

$$\tilde{\mathbf{\Delta}}_{\text{sym}}\mathbf{v}_1 = \tilde{\mathbf{D}}^{-1/2}\mathbf{\Delta}\tilde{\mathbf{D}}^{-1/2}\tilde{\mathbf{D}}^{1/2}\mathbf{v} = \tilde{\mathbf{D}}^{-1/2}\mathbf{\Delta v} = \mathbf{0}.$$

Therefore, $\mathbf{v}_1 = \tilde{\mathbf{D}}^{1/2}\mathbf{v}$ is an eigenvector of $\tilde{\mathbf{\Delta}}_{\text{sym}}$ associated with eigenvalue 0, which is then the smallest eigenvalue from the non-negativity of $\tilde{\mathbf{\Delta}}_{\text{sym}}$. Likewise, 0 can be proved to be the smallest eigenvalues of $\mathbf{\Delta}_{\text{sym}}$. □

Lemma 3. *Let $\beta_1 \leq \beta_2 \leq \cdots \leq \beta_n$ denote eigenvalues of $\mathbf{D}^{-1/2}\mathbf{A}\mathbf{D}^{-1/2}$ and $\alpha_1 \leq \alpha_2 \leq \cdots \leq \alpha_n$ be the eigenvalues of $\tilde{\mathbf{D}}^{-1/2}\mathbf{A}\tilde{\mathbf{D}}^{-1/2}$. Then,*

$$\alpha_1 \geq \frac{\max_i d_i}{\gamma + \max_i d_i}\beta_1, \qquad \alpha_n \leq \frac{\min_i d_i}{\gamma + \min_i d_i}. \qquad (2)$$

Proof. We have shown that 0 is an eigenvalue of $\mathbf{\Delta}_{\text{sym}}$. Since $\mathbf{D}^{-1/2}\mathbf{A}\mathbf{D}^{-1/2} = \mathbf{I} - \mathbf{\Delta}_{\text{sym}}$, then 1 is an eigenvalue of $\mathbf{D}^{-1/2}\mathbf{A}\mathbf{D}^{-1/2}$. More specifically, $\beta_n = 1$. In addition, by combining the fact that $\text{Tr}(\mathbf{D}^{-1/2}\mathbf{A}\mathbf{D}^{-1/2}) = 0 = \sum_i \beta_i$ with $\beta_n = 1$, we conclude that $\beta_1 < 0$.

By choosing $\mathbf{x}$ such that $\|\mathbf{x}\| = 1$ and $\mathbf{y} = \mathbf{D}^{1/2}\tilde{\mathbf{D}}^{-1/2}\mathbf{x}$, we have that $\|\mathbf{y}\|^2 = \sum_i \frac{d_i}{d_i+\gamma}x_i^2$ and $\frac{\min_i d_i}{\gamma+\min_i d_i} \leq \|\mathbf{y}\|^2 \leq \frac{\max_i d_i}{\gamma+\max_i d_i}$. Hence, we use the Rayleigh quotient to provide

[*] Equal contribution [1] Cornell University [2] Federal Institute of Ceara (Brazil). Correspondence to: Felix Wu <fw245@cornell.edu>, Tianyi Zhang <tz58@cornell.edu>.

Proceedings of the 36th International Conference on Machine Learning, Long Beach, California, PMLR 97, 2019. Copyright 2019 by the author(s).

a lower bound to α_1:

$$
\begin{aligned}
\alpha_1 &= \min_{\|\mathbf{x}\|=1} \left(\mathbf{x}^\top \tilde{\mathbf{D}}^{-1/2}\mathbf{A}\tilde{\mathbf{D}}^{-1/2}\mathbf{x}\right) \\
&= \min_{\|\mathbf{x}\|=1} \left(\mathbf{y}^\top \mathbf{D}^{-1/2}\mathbf{A}\mathbf{D}^{-1/2}\mathbf{y}\right) \text{ (by replacing variable)} \\
&= \min_{\|\mathbf{x}\|=1} \left(\frac{\mathbf{y}^\top \mathbf{D}^{-1/2}\mathbf{A}\mathbf{D}^{-1/2}\mathbf{y}}{\|\mathbf{y}\|^2}\|\mathbf{y}\|^2\right) \\
&\geq \min_{\|\mathbf{x}\|=1} \left(\frac{\mathbf{y}^\top \mathbf{D}^{-1/2}\mathbf{A}\mathbf{D}^{-1/2}\mathbf{y}}{\|\mathbf{y}\|^2}\right) \max_{\|\mathbf{x}\|=1}\left(\|\mathbf{y}\|^2\right) \\
&\quad (\because \min(AB) \geq \min(A)\max(B) \text{ if } \min(A) < 0, \forall B > 0, \\
&\quad \text{and } \min_{\|\mathbf{x}\|=1} \left(\frac{\mathbf{y}^\top \mathbf{D}^{-1/2}\mathbf{A}\mathbf{D}^{-1/2}\mathbf{y}}{\|\mathbf{y}\|^2}\right) = \beta_1 < 0) \\
&= \beta_1 \max_{\|\mathbf{x}\|=1} \|\mathbf{y}\|^2 \\
&\geq \frac{\max_i d_i}{\gamma + \max_i d_i}\beta_1.
\end{aligned}
$$

One may employ similar steps to prove the second inequality in Equation 2.

□

Proof of Theorem 1. Note that $\tilde{\mathbf{\Delta}}_{\text{sym}} = \mathbf{I} - \gamma\tilde{\mathbf{D}}^{-1} - \tilde{\mathbf{D}}^{-1/2}\mathbf{A}\tilde{\mathbf{D}}^{-1/2}$. Using the results in Lemma 3, we show that the largest eigenvalue $\tilde{\lambda}_n$ of $\tilde{\mathbf{\Delta}}_{\text{sym}}$ is

$$
\begin{aligned}
\tilde{\lambda}_n &= \max_{\|\mathbf{x}\|=1} \mathbf{x}^\top(\mathbf{I} - \gamma\tilde{\mathbf{D}}^{-1} - \tilde{\mathbf{D}}^{-1/2}\mathbf{A}\tilde{\mathbf{D}}^{-1/2})\mathbf{x} \\
&\leq 1 - \min_{\|\mathbf{x}\|=1} \gamma\mathbf{x}^\top\tilde{\mathbf{D}}^{-1}\mathbf{x} - \min_{\|\mathbf{x}\|=1} \mathbf{x}^\top\tilde{\mathbf{D}}^{-1/2}\mathbf{A}\tilde{\mathbf{D}}^{-1/2}\mathbf{x} \\
&= 1 - \frac{\gamma}{\gamma + \max_i d_i} - \alpha_1 \\
&\leq 1 - \frac{\gamma}{\gamma + \max_i d_i} - \frac{\max_i d_i}{\gamma + \max_i d_i}\beta_1 \\
&< 1 - \frac{\max_i d_i}{\gamma + \max_i d_i}\beta_1 \quad (\gamma > 0 \text{ and } \beta_1 < 0) \\
&< 1 - \beta_1 = \lambda_n
\end{aligned} \tag{3}
$$

□

B. Experiment Details

Node Classification. We empirically find that on Reddit dataset for SGC, it is crucial to normalize the features into zero mean and univariate.

Training Time Benchmarking. We hereby describe the experiment setup of Figure 3. Chen et al. (2018) benchmark the training time of FastGCN on CPU, and as a result, it is difficult to compare numerical values across reports. Moreover, we found the performance of FastGCN improved with a smaller early stopping window (10 epochs); therefore, we could decrease the model's training time. We provide the data underpinning Figure 3 in Table 1 and Table 2.

Table 1. Training time (seconds) of graph neural networks on Citation Networks. Numbers are averaged over 10 runs.

Models	Cora	Citeseer	Pubmed
GCN	0.49	0.59	8.31
GAT	63.10	118.10	121.74
FastGCN	2.47	3.96	1.77
GIN	2.09	4.47	26.15
LNet	15.02	49.16	266.47
AdaLNet	10.15	31.80	222.21
DGI	21.24	21.06	76.20
SGC	0.13	0.14	0.29

Table 2. Training time (seconds) on Reddit dataset.

Model	Time(s) ↓
SAGE-mean	78.54
SAGE-LSTM	486.53
SAGE-GCN	86.86
FastGCN	270.45
SGC	2.70

Text Classification. Yao et al. (2019) use one-hot features for the word and document nodes. In training SGC, we normalize the features to be between 0 and 1 **after propagation** and train with L-BFGS for 3 steps. We tune the only hyperparameter, weight decay, using hyperopt(Bergstra et al., 2015) for 60 iterations. Note that we cannot apply this feature normalization for TextGCN because the propagation cannot be precomputed.

Semi-supervised User Geolocation. We replace the 4-layer, highway-connection GCN with a 3rd degree propagation matrix ($K = 3$) SGC and use the same set of hyperparameters as Rahimi et al. (2018). All experiments on the GEOTEXT dataset are conducted on a single Nvidia GTX-1080Ti GPU while the ones on the TWITTER-NA and TWITTER-WORLD datasets are excuded with 10 cores of the Intel(R) Xeon(R) Silver 4114 CPU (2.20GHz). Instead of collapsing all linear transformations, we keep two of them which we find performing slightly better possibly due to . Despite of this subtle variation, the model is still linear.

Relation Extraction. We replace the 2-layer GCN with a 2nd degree propagation matrix ($K = 2$) SGC and remove the intermediate dropout. We keep other hyperparameters unchanged, including learning rate and regularization. Similar to Zhang et al. (2018), we report the best validation accuracy with early stopping.

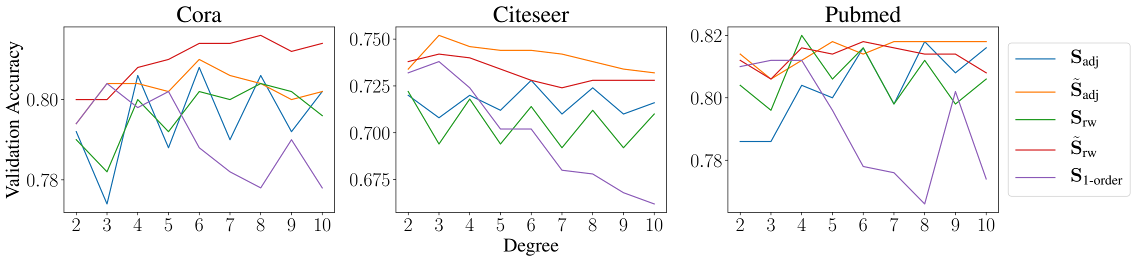

Figure 1. Validation accuracy with SGC using different propagation matrices.

Zero-shot Image Classification. We replace the 6-layer GCN (hidden size: 2048, 2048, 1024, 1024, 512, 2048) baseline with an 6-layer MLP (hidden size: 512, 512, 512, 1024, 1024, 2048) followed by a SGC with $K = 6$. Following (Wang et al., 2018), we only apply dropout to the output of SGC. Due to the slow evaluation of this task, we do not tune the dropout rate or other hyperparameters. Rather, we follow the GCNZ code and use learning rate of 0.001, weight decay of 0.0005, and dropout rate of 0.5. We also train the models with ADAM (Kingma & Ba, 2015) for 300 epochs.

C. Additional Experiments

Random Splits for Citation Networks. Possibly due to their limited size, the citation networks are known to be unstable. Accordingly, we conduct an additional 10 experiments on random splits of the training set while maintaining the same validation and test sets.

Table 3. Test accuracy (%) on citation networks (random splits). †We remove the outliers (accuracy $< 0.7/0.65/0.75$) when calculating their statistics due to high variance.

	Cora	Citeseer	Pubmed
Ours:			
GCN	80.53 ± 1.40	70.67 ± 2.90	77.09 ± 2.95
GIN	76.94 ± 1.24	66.56 ± 2.27	74.46 ± 2.19
LNet	$74.23 \pm 4.50^{\dagger}$	$67.26 \pm 0.81^{\dagger}$	$77.20 \pm 2.03^{\dagger}$
AdaLNet	$72.68 \pm 1.45^{\dagger}$	$71.04 \pm 0.95^{\dagger}$	$77.53 \pm 1.76^{\dagger}$
GAT	82.29 ± 1.16	72.6 ± 0.58	78.79 ± 1.41
SGC	80.62 ± 1.21	71.40 ± 3.92	77.02 ± 1.62

Propagation choice. We conduct an ablation study with different choices of propagation matrix, namely:

Normalized Adjacency: $\mathbf{S}_{\text{adj}} = \mathbf{D}^{-1/2}\mathbf{A}\mathbf{D}^{-1/2}$

Random Walk Adjacency $\mathbf{S}_{\text{rw}} = \mathbf{D}^{-1}\mathbf{A}$

Aug. Normalized Adjacency $\tilde{\mathbf{S}}_{\text{adj}} = \tilde{\mathbf{D}}^{-1/2}\tilde{\mathbf{A}}\tilde{\mathbf{D}}^{-1/2}$

Aug. Random Walk $\tilde{\mathbf{S}}_{\text{rw}} = \tilde{\mathbf{D}}^{-1}\tilde{\mathbf{A}}$

First-Order Cheby $\mathbf{S}_{\text{1-order}} = (\mathbf{I} + \mathbf{D}^{-1/2}\mathbf{A}\mathbf{D}^{-1/2})$

We investigate the effect of propagation steps $K \in \{2..10\}$ on validation set accuracy. We use hyperopt to tune L2-regularization and leave all other hyperparameters unchanged. Figure 1 depicts the validation results achieved by varying the degree of different propagation matrices.

We see that augmented propagation matrices (i.e. those with self-loops) attain higher accuracy and more stable performance across various propagation depths. Specifically, the accuracy of $\mathbf{S}_{\text{1-order}}$ tends to deteriorate as the power K increases, and this results suggests using large filter coefficients on low frequencies degrades SGC performance on semi-supervised tasks.

Another pattern is that odd powers of K cause a significant performance drop for the normalized adjacency and random walk propagation matrices. This demonstrates how odd powers of the un-augmented propagation matrix use negative filter coefficients on high frequency information. Adding self-loops to the propagation matrix shrinks the spectrum such that the largest eigenvalues decrease from ≈ 2 to ≈ 1.5 on the citation network datasets. By effectively shrinking the spectrum, the effect of negative filter coefficients on high frequencies is minimized, and as a result, using odd-powers of K does not degrade the performance of augmented propagation matrices. For non-augmented propagation matrices — where the largest eigenvalue is approximately 2 — negative coefficients significantly distort the signal, which leads to decreased accuracy. Therefore, adding self-loops constructs a better domain in which fixed filters can operate.

Data amount. We also investigated the effect of training dataset size on accuracy. As demonstrated in Table 4, SGC continues to perform similarly to GCN as the training dataset size is reduced, and even outperforms GCN when there are fewer than 5 training samples. We reason this study demonstrates SGC has at least the same modeling capacity as GCN.

# Training Samples	SGC	GCN
1	33.16	32.94
5	63.74	60.68
10	72.04	71.46
20	80.30	80.16
40	85.56	85.38
80	90.08	90.44

Table 4. Validation Accuracy (%) when SGC and GCN are trained with different amounts of data on Cora. The validation accuracy is averaged over 10 random training splits such that each class has the same number of training examples.

References

Bergstra, J., Komer, B., Eliasmith, C., Yamins, D., and Cox, D. D. Hyperopt: A python library for optimizing the hyperparameters of machine learning algorithms. *Computational Science & Discovery*, 8(1), 2015.

Chen, J., Ma, T., and Xiao, C. FastGCN: Fast Learning with Graph Convolutional Networks via Importance Sampling. In *International Conference on Learning Representations (ICLR'2018)*, 2018.

Kingma, D. P. and Ba, J. Adam: A method for stochastic optimization. In *International Conference on Learning Representations (ICLR'2015)*, 2015.

Rahimi, S., Cohn, T., and Baldwin, T. Semi-supervised user geolocation via graph convolutional networks. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 2009–2019. Association for Computational Linguistics, 2018.

Wang, X., Ye, Y., and Gupta, A. Zero-shot recognition via semantic embeddings and knowledge graphs. In *Computer Vision and Pattern Recognition (CVPR)*, pp. 6857–6866. IEEE Computer Society, 2018.

Yao, L., Mao, C., and Luo, Y. Graph convolutional networks for text classification. In *Proceedings of the 33rd AAAI Conference on Artificial Intelligence (AAAI'19)*, 2019.

Zhang, Y., Qi, P., and Manning, C. D. Graph convolution over pruned dependency trees improves relation extraction. In *Empirical Methods in Natural Language Processing (EMNLP)*, 2018.