Simplifying Graph Convolutional Networks

Elias Castro, Manolis Huerta-Stylianou, Tuni Le, Josue Ortiz

Cornell University

Cornell Bowers C·IS
College of Computing
and Information Science

Introduction

Graph Convolutional Networks (GCNs) emerged as the most widely used models for learning graph representations.

However, drawing inspiration from deep learning methods has caused them to inherit unnecessarily complex computational processes.

Simple Graph Convolution (SGC) (Wu et al., 2019) reduces computational complexity by removing nonlinearities and collapsing weight matrices across consecutive layers, allowing the graph propagation step to be precomputed once. This way, the model only needs to train a simple logistic regression layer, significantly reducing both memory usage and computation time. This is done without compromising accuracy.

Using prominent citation networks as datasets, we reproduce the results from Wu et al. and demonstrate that SGC maintains accuracy while achieving a substantial increase in efficiency, improving performance by two orders of magnitude compared to the traditional GCN (Kipf & Welling, 2017).

Methodology

Datasets: Cora, PubMed, Citeseer

We reproduced a GCN and an SGC and trained them on these three datasets ten times for each model and dataset, with the aim of comparing accuracy with each other and those of the original papers.

Models:

GCN (Kipf & Welling, 2017)

Two-layer architecture (Captures local neighborhood structure via stacked propagation):

- Layer 1: Graph convolution → ReLU → 50% dropout.
- Layer 2: Graph convolution → raw logits.

SGC (Wu et al., 2019)

Simplified GCN by collapsing k-layer propagation into a single linear transformation.

No nonlinearities or dropout.

Softmax applied externally during training.

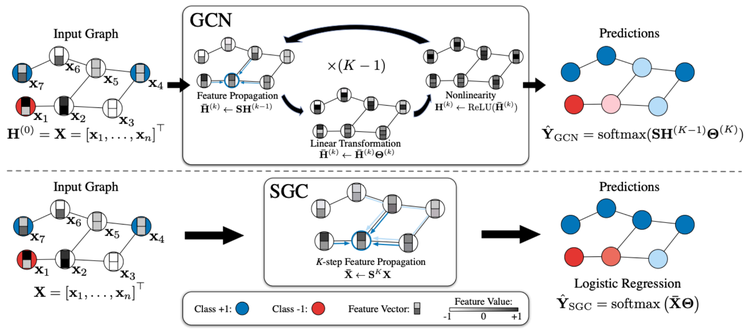
Training Configuration:

GCN

Hidden units: 16; Dropout: 50%; Learning rate: 0.01; Optimizer: Adam; Epochs: 200; Loss: Cross-Entropy.

SGC

One linear layer; No dropout; Learning rate: 0.2; Optimizer: Adam; Epochs: 100; Loss: Cross-Entropy.



Schematic layout of a GCN versus a SGC as shown in Wu et al.

Results

Model Performance on Each Dataset

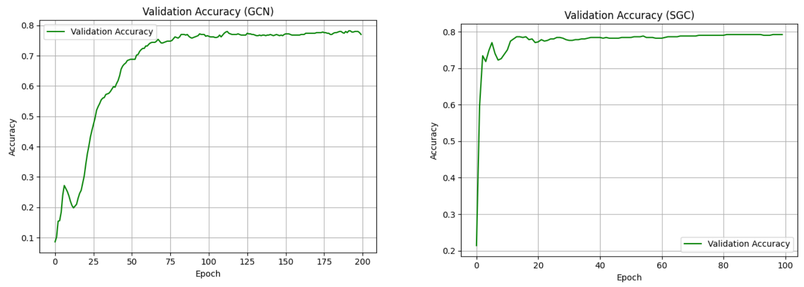
After taking the mean and standard deviations of our ten resulting accuracies, we compare the training accuracies of our Graph Convolutional Network (GCN) and Simplified Graph Convolution (SGC) implementations to the results reported in their respective original papers:

	Cora	Citeseer	Pubmed
GCN (original experiment)	81.5	70.3	79.0
GCN (their experiment)	81.4 ± 0.4	70.9 ± 0.5	79.0 ± 0.4
GCN (our experiment)	80.7 ± 0.8	70.9 ± 0.7	79.0 ± 0.5
SGC (their experiment)	81.0 ± 0.0	71.9 ± 0.1	78.9 ± 0.0
SGC (our experiment)	80.6 ± 0.1	68.3 ± 0.1	77.9 ± 0.1

Evaluation

Despite the slightly lower accuracy in our SGC implementation compared to expected performance benchmarks, our results show consistent performance across both GCN and SGC models. Notably, our GCN model even slightly outperforms the original.

These findings support the conclusion that our implementations maintain strong predictive performance without sacrificing efficiency, reinforcing the viability of simplified architectures like SGC in practical scenarios.



Validation Accuracy over 200 epochs for our SGCs on the Cora dataset.

Conclusion

This project demonstrates that Simple Graph Convolution (SGC) achieves accuracy comparable to traditional Graph Convolutional Networks (GCNs) on the Cora citation dataset, while offering significant computational efficiency gains.

Our reimplementation supports the findings of Wu et al. (2019), indicating that removing nonlinearities and collapsing weight matrices across layers can be an effective strategy for certain graph learning tasks.

Future Work

Currently, our SGC succeeds only in limited settings under the assumption that data isn't noisy or incomplete. We aim to address these limitations with the following future implementations:

Extend datasets outside of citation networks – some examples include social networks, protein interaction graphs, and recommendation systems.

Incorporate graph structure learning – refine graph topology during training.

References

Wu, Felix, et al. "Simplifying Graph Convolutional Networks." Arxiv, 20 June 2019, arxiv.org/pdf/1902.07153.

Thomas, Kipf N, et al. "Semi-Supervised Classification with Graph Convolutional Networks." Arxiv, 22 Feb. 2017, arxiv.org/pdf/1609.02907.pdf.