PROPOSAL: Sparse Self-Attention in Graph Convolutional Networks for Improved Structure Learning Runtime

Anton Chen [*1] **Xiang Xi Chen** [*1]

Abstract

Graph Convolutional Networks (GCNs) are increasingly popular architectures for representation learning and downstream tasks on graph-structured data. While the majority of modern GCNs face difficulty learning low-homophily graphs, a recent approach (GCN-SA) successfully captures long-range dependencies using multi-head self-attention. However GCN-SA is self-admittedly limited by the poor runtime scaling of its self-attention mechanism. We propose the use of sparse self-attention mechanisms (e.g. Big Bird, Performer) to replace GCN-SA's attention mechanisms to reduce SA runtime from quadratic to linear. As noted in the mentioned sparse SA literature, we expect minimal performance degradation while improving GCN-SA's runtime.

1. Context and Problem

1.1. Graph-Structured Data and GCNs

It's no secret that many real-world problems involve graph-structured data, including classification on social networks, academic paper citations, webpage links, and so on. Graph Convolutional Networks (GCNs) have evolved as generalizations of CNNs, and are considered the most promising architecture to handle graph data, having already been used successfully in numerous applications (Hamilton et al., 2017).

1.2. Non-Homophilous Graphs Pose Problems

One glaring issue that modern GCNs face is the task of learning low-homophily graphs (Fan et al., 2023).

[*]Equal contribution [1]Department of Computer Science, University of British Columbia, Vancouver, B.C., Canada. Correspondence to: Anton Chen <contact@antonchen.ca>, Xiang Xi Chen <xiangxi.chen.ca@gmail.com>.

Advanced Machine Learning, CPSC 440, 2024W2.

Homophily in graph data refers to the tendency for nodes with similar features to be closely connected (Li et al., 2023). For instance, in a professional network, people tend to be connected with similar individuals (e.g. similar industry, position, years of experience, etc.), illustrating an example of a high-homophily graph. An example of a low-homophily graph is a subway system where stations serve as a hub for diverse individuals with varying demographics, interests, and use for the subway.

Modern GCNs tend to perform poorly on low-homophily graph data, including Geom-GCN (Pei et al., 2020) and H_2GCN (Zhu et al., 2020). This is due to such approaches relying on feature representations that only consider adjacent neighbor nodes, failing to capture long-range correlations (Jiang et al., 2024). Jiang et al. takes this into account in their newly proposed model.

1.3. GCN-SA, Full Self-Attention, and Poor Runtime

Jiang et al. (Martch 2024) introduces GCN-SA, an architecture which leverages multi-head self-attention (MHSA) to effectively learn node and edge embeddings in both low and high-homophily graphs (Jiang et al., 2024).

What sparks our interest the most about GCN-SA is its creative use of a self-attention mechanism: MHSA is used to learn a new adjacency matrix where self-attention identifies highly-correlated nodes, and transforms the adjacency matrix such that they become neighbors. In particular, long-distance correlations can be uncovered, resulting in better performance on low-homophily datasets.

A significant limitation of GCN-SA mentioned by (Jiang et al., 2024) is the poor runtime complexity caused by its full self-attention mechanism, which necessitates n^2 pairwise operations for n nodes. Our main motivation for this project is to reduce the "excessive computational burden", which becomes particularly problematic when dealing with incredibly large non-homophilous graphs in practice (Lim et al., 2021).

1.4. Sparse Self Attention: Big Bird, Performers, etc.

Outside of graph learning, researchers have found great success in efficient long-range attention with the rise of sparse self-attention mechanisms (Zaheer et al., 2020; Choromanski et al., 2020; Beltagy et al., 2020). Rather than computing all pairwise similarity scores, sparse self-attention mechanisms compute a small subset of similarity scores, yielding a sparse score matrix in a fraction of the time. In fact, self-attention runtime complexity is reduced from *quadratic* runtime to *linear* runtime.

The fact that the resulting score matrix is sparse is not an issue — it is actually desired. A sparse score matrix corresponds to less edge connections in the reconnected adjacency matrix. Jiang et al. even intend to ensure the reconnected adjacency is sparse, with a k-NN and minimum-threshold approach to selecting attention. Linear attention will do this all for us — and faster.

Sparse attention methods such as Performers, a linear attention framework that approximate the standard attention matrix into lower-rank randomized matrices (Choromanski et al., 2020), and Big Bird, a framework that does not require prerequisite knowledge about source data structures (Zaheer et al., 2020), have successfully acted as generic drop-in replacements for full self-attention, showing promise in new domains without existing domain knowledge. This motivates the use of such methods in GCN-SA for graph learning.

2. Proposed Plan, Procedure, and Experiments

Our high-level idea is to modify GCN-SA's self-attention-based adjacency matrix transforming mechanism with various sparse self-attention mechanisms. We'll focus on classification accuracy and runtime of the modified models against baseline GCN-SA.

More specifically, our plan includes:

1. Technical overview of GCN-SA's full self-attention mechanism, focusing on the mathematics.
2. Technical overview of sparse (and linear) self-attention mechanisms.
3. Creating multiple variations of GCN-SA, each with linear self-attention layers (e.g. one model with Big Bird layers, one model with Performer layers, etc.)
4. Perform training/validation/testing on one of the 8 datasets, similar to Jiang et al.'s method. We'll also experiment with varying hyperparameters via cross-validation, e.g. number of heads in self-attention.
5. Repeat on all 8 datasets of varying degrees of homophily,
6. Introduce more linear self-attention mechanisms if time permits.
7. Compare runtime and classification accuracy. Draw conclusions, make reflections, etc.

2.1. Available Source Code and Datasets

Source code for GCN-SA, Big Bird, and Performers is available on GitHub. After some preliminary investigation into the codebases, GCN-SA implements self-attention with in a custom PyTorch module. Big Bird and Performers expose linear self-attention as layers, so minimal "model surgery" is required.

We'll be using the 8 open graph datasets used in Jiang et al.'s study. Note that these datasets have varying degrees of homophily, so we are especially interested in performance on the low-homophily datasets.

2.2. Time Constraints

To vary the scope of the project, we may reduce the number of sparse self-attention techniques used, or the number of datasets trained on.

3. Potential Impact and Contribution

We'd like to see significant runtime improvements with similar performance for these sparse self-attention models. If successful, we can greatly reduce the time it takes to train large and non-homophilous graphs. As mentioned previously, this has widespread applications in social network data, global trade data, etc and can make meaningful real world contributions. Improved training runtime can open up the door for possibilities including better downstream accuracy, or the ability to use larger datasets.

References

Beltagy, I., Peters, M. E., and Cohan, A. Longformer: The long-document transformer. *arXiv preprint arXiv:2004.05150*, 2020.

Choromanski, K., Likhosherstov, V., Dohan, D., Song, X., Gane, A., Sarlos, T., Hawkins, P., Davis, J., Mohiuddin, A., Kaiser, L., et al. Rethinking attention with performers. *arXiv preprint arXiv:2009.14794*, 2020.

Fan, X., Gong, M., and Wu, Y. Markov clustering regularized multi-hop graph neural network. *Pattern Recognition*, 139:109518, 2023.

Hamilton, W. L., Ying, R., and Leskovec, J. Representation

learning on graphs: Methods and applications. *arXiv preprint arXiv:1709.05584*, 2017.

Jiang, M., Liu, G., Su, Y., and Wu, X. Self-attention empowered graph convolutional network for structure learning and node embedding. *arXiv preprint arXiv:2403.03465*, 2024.

Li, W.-Z., Wang, C.-D., Xiong, H., and Lai, J.-H. Homogcl: Rethinking homophily in graph contrastive learning. In *Proceedings of the 29th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, pp. 1341–1352, 2023.

Lim, D., Hohne, F., Li, X., Huang, S. L., Gupta, V., Bhalerao, O., and Lim, S. N. Large scale learning on non-homophilous graphs: New benchmarks and strong simple methods. *Advances in Neural Information Processing Systems*, 34:20887–20902, 2021.

Pei, H., Wei, B., Chang, K. C.-C., Lei, Y., and Yang, B. Geom-gcn: Geometric graph convolutional networks. *arXiv preprint arXiv:2002.05287*, 2020.

Zaheer, M., Guruganesh, G., Dubey, K. A., Ainslie, J., Alberti, C., Ontanon, S., Pham, P., Ravula, A., Wang, Q., Yang, L., et al. Big bird: Transformers for longer sequences. *Advances in neural information processing systems*, 33:17283–17297, 2020.

Zhu, J., Yan, Y., Zhao, L., Heimann, M., Akoglu, L., and Koutra, D. Beyond homophily in graph neural networks: Current limitations and effective designs. *Advances in neural information processing systems*, 33:7793–7804, 2020.