Research Direction Proposal

Zheng Huang

September 2022

Graph Neural Networks (GNNs) has emerged as a powerful machine learning model for exploring the massive amount of graph data collected in a single machine. However, in real-world scenarios, due to data security and user privacy, the graph data stored in each device is limited and often has bias. Even though federated GNN models are proposed to solve the data isolation and heterogeneity problem with pseudo interacted items [1] and missing neighbor generation [2], these models lack explainability (e.g., the prediction of these models is not traceable). Thus, a trustworthy federated GNN model is expected.

We aim to train an explainable GNN model orchestrated by a central server jointly with sub-graphs in each client. To improve the expressive ability of GNN with heterogenous data, inspired by LAGNN [3], we could use augmentation strategies. Regarding the graph in the client with a limited number of edges and nodes, we can add more neighbors and features based on the local neighbors with a generative model like the one used in LAGNN. Thus, we can ensure that each client shares a similar graph size. Moreover, to improve explainability, we want to focus on structural explainability [4]. After we have the augmented graphs, we feed them into a structural explainer. The explainer aims to train a fairer GNN by masking bias edges (edge set s1) while identifying edges that increase fairness (edge set s2) in the augmented graph. The two sets, s1 and s2, can help us understand which edges are not necessary when making the prediction and which edges contribute fairness to the final prediction.

From the perspective of the Federated Learning (FL) system, another challenge that biases the trained model is slow-responding clients. In traditional synchronous FL systems [5], due to the data heterogeneity, some clients take a longer time to complete each training round. The system then discards these clients, which biases the model training. Inspired by PAPAYA [6], I believe an asynchronous FL system can be utilized to train our GNN model. The system makes sure weighs can be aggregated by the server as soon as they are ready. Thus, without discarding updates from slow clients, we can achieve a fairer model.

References

[1] C. Wu, F. Wu, Y. Cao, Y. Huang, and X. Xie, “Fedgnn: Federated graph neural network for privacy-preserving recommendation,” *arXiv preprint arXiv:2102.04925*, 2021.

[2] K. Zhang, C. Yang, X. Li, L. Sun, and S. M. Yiu, “Subgraph federated learning with missing neighbor generation,” *Advances in Neural Information Processing Systems*, vol. 34, pp. 6671–6682, 2021.

[3] S. Liu, R. Ying, H. Dong, L. Li, T. Xu, Y. Rong, P. Zhao, J. Huang, and D. Wu, “Local augmentation for graph neural networks,” in *International Conference on Machine Learning*, pp. 14054–14072, PMLR, 2022.

[4] Y. Dong, S. Wang, Y. Wang, T. Derr, and J. Li, “On structural explanation of bias in graph neural networks,” in *Proceedings of the 28th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, pp. 316–326, 2022.

[5] J. H. Bell, K. A. Bonawitz, A. Gascón, T. Lepoint, and M. Raykova, “Secure single-server aggregation with (poly) logarithmic overhead,” in *Proceedings of the 2020 ACM SIGSAC Conference on Computer and Communications Security*, pp. 1253–1269, 2020.

[6] D. Huba, J. Nguyen, K. Malik, R. Zhu, M. Rabbat, A. Yousefpour, C.-J. Wu, H. Zhan, P. Ustinov, H. Srinivas, *et al.*, “Papaya: Practical, private, and scalable federated learning,” *Proceedings of Machine Learning and Systems*, vol. 4, pp. 814–832, 2022.