Relational Deep Learning with Graph Transformers: Exploring Local and Global Message Passing

Ignacio Cuñado Barral[1] supervised by Prof. Kubilay Atasu[1], Çağrı Bilgi[1]

[1]EEMCS, Delft University of Technology, The Netherlands

Abstract

Graph Transformers have played a key role in the latest graph learning developments. However their performance in Relational Deep Learning remains largely unexplored. We propose adaptations to two Graph Transformer models implementing local message passing and global attention and evaluate them on RelBench, a set of comprehensive RDL benchmarks. We show that local message passing has a lower complexity, requiring less memory and training time, and outperforms global attention. We demonstrate that our implementation can achieve state of the art results on node classification and regression tasks.

Introduction

Graph Neural Networks learn meaningful representation and patterns from graph-structured data, and rely heavily on **message passing** (MP). **Graph Transformers** were introduced as an extension, adapting the concept of **attention** to graphs [1]. Relational deep learning (RDL) aims to learn data from tables in a database without doing a feature engineering step, saving time and cost [2]. We implement FraudGT [3] for local MP and Graphormer [4] for global attention to answer whether one outperforms the other on RelBench [5] tasks?

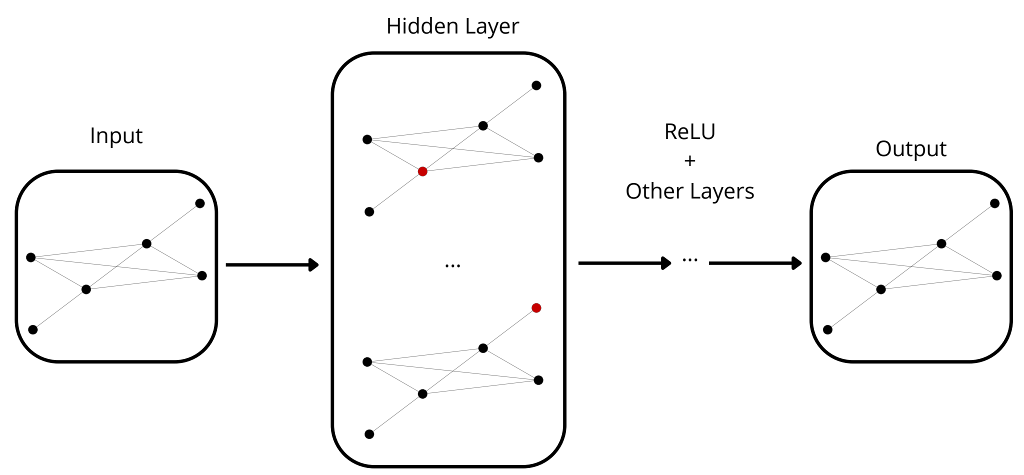

Figure 1. Graph Neural Network Architecture

Research Questions

- How do **runtime** and **memory usage** differ between global and local message-passing transformers as graph size scales?
- Does a global message-passing scheme **simulate or strictly dominate** the representational power of purely local message-passing in graph transformers?
- On RelBench node classification benchmarks, which message-passing scheme achieves **higher accuracy** (ROC-AUC)?
- For RelBench graph-scoring (regression) tasks, how do global and local message-passing architectures compare in terms of mean absolute error (MAE)?

Background

Relational Deep Learning (RDL) represents databases as heterogeneous graphs — rows as nodes, columns as features, and foreign key links as edges. Graph learning method can then be used for predictive tasks.

Graph Transformers use different attention biases to learn the graph's structure and node positions. In each Transformer layer, multi-head attention is performed.

Attention mechanisms can perform **local attention**, in which each node attends only the nodes in its neighbourhood; or **global attention**, where each node attends to every other node. Global models are obviously more expensive than local message passing, with a layer complexity of $\mathcal{O}(n^2)$, since now every node has to attend every other node in the graph.

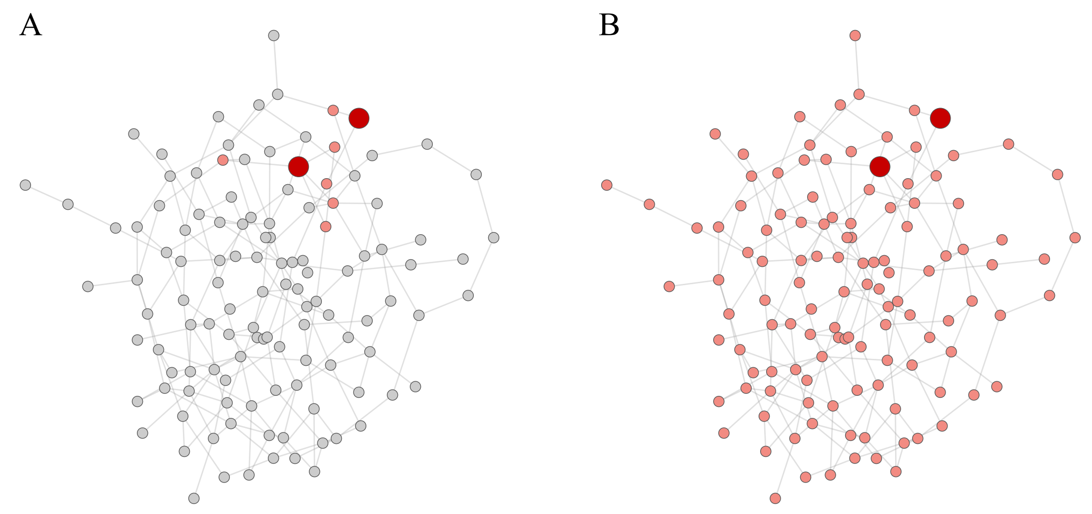

Figure 2. Local and global attention mechanisms when computing attention between two nodes. **A** (left) shows local message passing, **B** (right) is global attention.

Methodology

1. Design and adapt attention mechanisms and implement the transformers (FraudGT for local message passing and Graphormer for global message passing).
2. Evaluate models on 6 classification and 5 regression tasks from RelBench.
3. Tune hyperparameters via Bayesian search.
4. For each task, run 5 train-test loops with different random seeds. Compute the average and standard deviation.

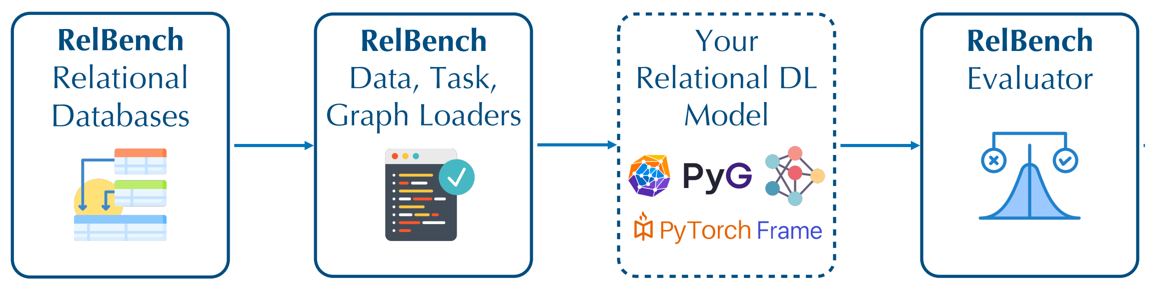

Figure 3. RelBench evaluation pipeline: starting from a relational database, loaders prepare graph inputs for our custom models which are then evaluated.

Results and Discussion

We present a summary of results, which show that Local message passing models like FraudGT can **generally outperform** GNN-based Relational Deep Learning implementations. Their relatively **low complexity and memory usage**, as well as fast training times make it a good option for a variety of scenarios.

Graph attention mechanisms **cannot simulate the expressive power of purely local message passing Transformers**. Even with reduced batch sizes and a limited number of neighbours, training global models demands the maximum memory capacity of even the latest GPUs.

This forces training models with a smaller batch size; such that, when increased layer complexity is taken into account, training and inference times are even slower.

Dataset / Task	RDL	Local MP	Global Attn
F1 (driver-top3)	75.54±0.63	**82.99±0.87**	76.67±2.80
HM (user-churn)	69.88±0.21	**70.30±0.30**	67.49±0.03
Trial (study-outcome)	68.60±1.01	**69.28±0.32**	67.42±0.59
Avito (user-visits)	**66.20±0.10**	64.92±0.20	63.72±0.44

Table 1. Node classification results (ROC-AUC, higher is better). Mean±SD over 5 runs; best scores are in bold.

Dataset / Task	RDL	Local MP	Global Attn
F1 (driver-position)	4.022±0.119	**3.925±0.062**	4.025±0.085
HM (item-sales)	0.056±0.000	**0.052±0.003**	0.076±0.000
Trial (site-success)	0.400±0.020	**0.376±0.024**	0.450±0.013
Trial (study-adverse)	44.473±0.209	**43.439±0.771**	47.271±0.802

Table 2. Node regression results (MAE, lower is better). Mean±SD over 5 runs; best scores are bold.

References

[1] S. Yun, M. Jeong, R. Kim, J. Kang, and H. J. Kim, *Graph transformer networks*, 2020. arXiv: `1911.06455 [cs.LG]`.

[2] M. Fey *et al.*, *Relational deep learning: Graph representation learning on relational databases*, 2023. arXiv: `2312.04615 [cs.LG]`.

[3] J. Lin, X. Guo, Y. Zhu, S. Mitchell, E. Altman, and J. Shun, "Fraudgt: A simple, effective, and efficient graph transformer for financial fraud detection," in *Proceedings of the 5th ACM International Conference on AI in Finance*, 2024, pp. 292–300. DOI: `10.1145/3677052.3698648`.

[4] C. Ying *et al.*, "Do transformers really perform badly for graph representation?" In *Advances in Neural Information Processing Systems*, M. Ranzato, A. Beygelzimer, Y. Dauphin, P. Liang, and J. W. Vaughan, Eds., vol. 34, Curran Associates, Inc., 2021, pp. 28 877–28 888.

[5] J. Robinson *et al.*, *Relbench: A benchmark for deep learning on relational databases*, 2024. arXiv: `2407.20060 [cs.LG]`.