Graph Neural Networks go grammatical

Jason Piquenot[1], Aldo Moscatelli[1], Maxime Bérar[1], Pierre Héroux[1], Jean-Yves Ramel[2], Romain Raveaux[2], and Sébastien Adam[1]

[1]LITIS Lab, University of Rouen Normandy, France
[2]LIFAT Lab, University of Tours, France

Abstract

In the last few years, characterising the expressive power of Graph Neural Networks (GNNs) has become a major concern in order to theoretically rank the plethora of existing models and to design new provably powerful GNNs [1]. In this field, the Weisfeiler-Lehman hierarchy, based on the eponymous polynomial-time test [2], is the most common way to characterise architectures. However, most of the time 3-WL and 1-WL are only bounds of expressive power. A recent way to refine this WL hierarchy is to study the substructures GNNs can count in a graph [3]. A complementary way of measuring the expressive power of a model consists in assessing its spectral ability, in order to take into account the effect of the model on the signal handled by the nodes [4].

Taking into consideration the three characterization aspects mentioned above, this talk proposes a new GNN design strategy and a new model called Grammatical Graph Neural Network (G^2N^2) resulting from this strategy. Our strategy relies on the MATLANG language introduced in [5] and more particularly on the fragments of MATLANG called $\mathrm{ML}(\mathcal{L}_1)$ and $\mathrm{ML}(\mathcal{L}_3)$, shown to be as expressive as 1-WL and 3-WL in [6]. Starting from the operations sets $\mathcal{L}_1$ and $\mathcal{L}_3$, we propose to build Context-Free Grammars (CFG) able to generate $\mathrm{ML}(\mathcal{L}_1)$ and $\mathrm{ML}(\mathcal{L}_3)$. Since the amount of possible operations in the corresponding CFGs is pretty high, those CFGs are reduced, keeping the equivalence with 1-WL and 3-WL. From the variables of those reduced CFGs, GNN inputs can easily be deduced. Then, the rules of the CFGs determine the GNN layers update rules and the readout functions. A direct benefit of this design strategy is that GNN abilities can be deduced from the study of the language derived from the CFG.

Our strategy provably ensures that our G^2N^2 model is (i) **exactly as expressive as 3-WL** since it inherits expressive power of $\mathrm{ML}(\mathcal{L}_3)$, (ii) **able to count important substructures both at node-, graph- and edge-levels**, surpassing the counting abilities of existing MPNNs and subgraph MPNNs and (iii) **able to approximate low-pass, high-pass and band-pass filters** in the spectral domain while most models and in particular PPGN cannot experimentally approximate band-pass filters. These theoretical results are confirmed by numerous experiments on various well-known dedicated graph datasets.

References

[1] Christopher Morris, Martin Ritzert, Matthias Fey, William L. Hamilton, Jan Eric Lenssen, Gaurav Rattan, and Martin Grohe. Weisfeiler and lehman go neural: Higher-order graph neural networks. In *Proceedings of the Thirty-Third AAAI Conference on Artificial Intelligence*, AAAI'19. AAAI Press, 2019.

[2] AA Lehman and Boris Weisfeiler. A reduction of a graph to a canonical form and an algebra arising during this reduction. *Nauchno-Technicheskaya Informatsiya*, 2(9):12–16, 1968.

[3] Zhengdao Chen, Lei Chen, Soledad Villar, and Joan Bruna. Can graph neural networks count substructures? *Advances in neural information processing systems*, 33:10383–10395, 2020.

[4] Muhammet Balcilar, Guillaume Renton, Pierre Héroux, Benoit Gaüzère, Sébastien Adam, and Paul Honeine. Analyzing the expressive power of graph neural networks in a spectral perspective. In *International Conference on Learning Representations*, 2021.

[5] Robert Brijder, Floris Geerts, Jan Van den Bussche, and Timmy Weerwag. On the expressive power of query languages for matrices. *ACM Trans. Database Syst.*, 44(4):15:1–15:31, 2019.

[6] F Geerts. On the expressive power of linear algebra on graphs. *Theory of Computing Systems*, Oct 2020.