Breaking the Limits of Message Passing Graph Neural Networks

Muhammet Balcilar, Pierre Héroux, Benoit Gaüzère, Pascal Vasseur, Sébastien Adam, Paul Honeine

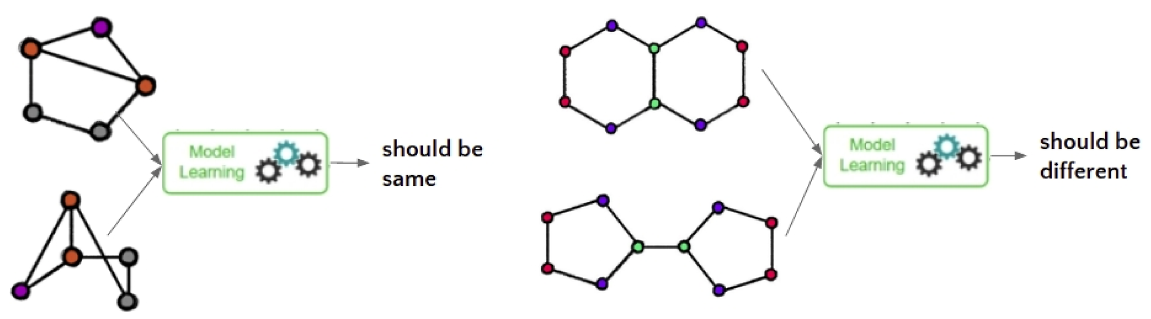
Normandy University, LITIS Lab, University of Rouen Normandy, INSA Rouen Normandie, InterDigital Inc, Université de Picardie Jules Verne, France

Contributions

- A new Message Passing (Graph) Neural Network (MPNN) that has 3-WL expressive power without feature engineering, while keeping complexity linear.
- By translating the insights of MATLANG to the GNN world, we show the expressive power of some GNN models and what we need in order to count some graphlets.

Introduction

- Universality of the Graph Neural Networks (GNN) depends on permutational invariance and ability to produce different outputs for non-isomorphic graphs.



- Expressive power of GNNs were evaluated using equivalence of Weisfeiler-Lehman test order.
- MPNN that has linear time and memory complexity is known to be 1-WL equivalent GNN.
- 1-WL equivalent methods cannot count some substructures in the graph which is crucial for many graph learning problems.
- For more expressive power GNN, we need to mimic higher order WL-test in exchange of increasing complexity exponentially.
- PPGN is the best known 3-WL equivalent GNN which has $\mathcal{O}(n^3)$ time and $\mathcal{O}(n^2)$ memory complexity.

Message Passing Neural Networks

- Both Spectral and Spatial MPNNs are generalized by:

$$H^{(l+1)} = \sigma\Big(\sum_s C^{(s)} H^{(l)} W^{(l,s)}\Big) \qquad (1)$$

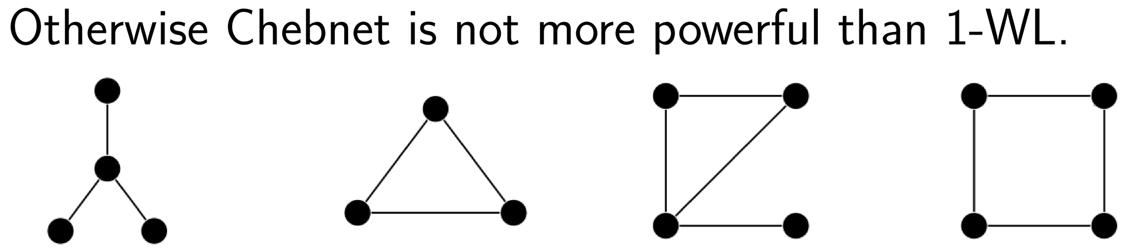
- Spatial GNNs are shown by convolution supports $C^{(s)}$.
- Spectral GNNs designed by frequency response $\Phi_s(\boldsymbol{\lambda})$, that can define supports by $C^{(s)} = U(\Phi_s(\boldsymbol{\lambda}))U^\top$.
- As long as $C^{(s)}$ are sparse, MPNN has linear time and memory complexity wrt number of nodes.

Characterization of Weisfeiler-Lehman

- 1-WL=2-WL <3-WL<4-WL<......<k-WL
- Recently, Matrix Language MATLANG was proposed to charactize WL test.
- **Definition1** $ML(\mathcal{L})$ is a matrix language with an allowed operation set $\mathcal{L} = \{op_1, \dots op_n\}$, where $op_i \in \{., +, ^\top, diag, tr, \mathbf{1}, \odot, \times, f\}$.
- **Definition2** $e(X) \in \mathbb{R}$ is a sentence in $ML(\mathcal{L})$ if it consists of any possible consecutive operations in $\mathcal{L}$, operating on a given matrix X and resulting in a scalar.
- **Remark1** Two adjacency matrices are indistinguishable by the 1-WL test if and only if $e(A_G) = e(A_H)$ for all $e \in \mathcal{L}_1$ with $\mathcal{L}_1 = \{., ^\top, \mathbf{1}, diag\}$.
- **Remark2** $ML(\mathcal{L}_2)$ with $\mathcal{L}_2 = \{., ^\top, \mathbf{1}, diag, tr\}$ is strictly more powerful than the 1-WL test, but less powerful than the 3-WL test.
- **Remark3** Two adjacencies are indistinguishable by the 3-WL test if and only if they are indistinguishable by any sentence in $\mathcal{L}_3 = \{., ^\top, \mathbf{1}, diag, tr, \odot\}$
- **Remark4** Enriching the operation set to $\mathcal{L}^+ = \mathcal{L} \cup \{+, \times, f\}$ does not improve the expressive power of the language.

How Powerful are MPNNs?

- **Theorem1** MPNNs such as GCN, GAT, GraphSage, GIN cannot go further than operations in $\mathcal{L}_1^+$. Thus, they are not more powerful than the 1-WL test.
- **Theorem2** Chebnet is more powerful than the 1-WL test if the Laplacian maximum eigenvalues of the non-regular graphs to be compared are not the same. Otherwise Chebnet is not more powerful than 1-WL.



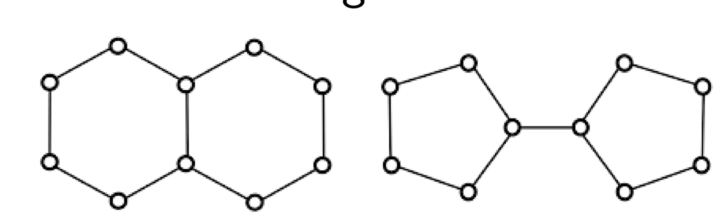
- **Theorem 3,4,5** 3-star graphlets can be counted by sentences in $\mathcal{L}_1^+$, Triangle and 4-cycle graphlets can be counted by sentences in $\mathcal{L}_2^+$, Tailed triangle graphlets can be counted by sentences in $\mathcal{L}_3^+$.

MPNN Beyond 1-WL

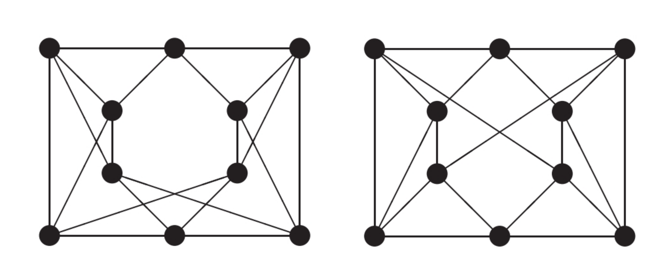
- MATLANG says any GNN that can produce all possible sentences from $\mathcal{L}_1 = \{., ^\top, \mathbf{1}, diag\}$ has exactly 1-WL expressive power.
- **Theorem6 GNNML1** given in the equation can produce every possible sentences in $ML(\mathcal{L}_1)$. Thus, GNNML1 is exactly as powerful as the 1-WL test.

$$H^{(l+1)}{=}\sigma\Big(H^{(l)}W^{(l,1)}{+}AH^{(l)}W^{(l,2)}{+}H^{(l)}W^{(l,3)}{\odot}H^{(l)}W^{(l,4)}\Big) \qquad (2)$$

- MATLANG says GNN should also produce sentences with $\{tr, \odot\}$ operations to have 3-WL test power.
- Since $tr(A_G^5) \neq tr(A_H^5)$ for pair of graphs in the figure, trace operation can distinguish them.



- Trace is not helpful for cospectral graph pairs. But $\odot$ operation helps. For instance this sentence for given graphs. $\mathbf{1}^\top((A_G \odot A_G^2)^2\mathbf{1})^2 \neq \mathbf{1}^\top((A_H \odot A_H^2)^2\mathbf{1})^2$



- MPNN does not keep any power of adjacency explicitly. It can not apply trace or elementwise multiplication.
- We proposed to design convolution support in spectral domain in preprocessing step, to be able to learn necessary power of adjacency.
- **Theorem7** A convolution support given by

$$C'^{(s)} = Udiag(\Phi_s(\lambda))U^\top,$$

where $\Phi_s(\lambda) = exp(-b(\lambda - f_s)^2)$, $f_s \in [\lambda_{min}, \lambda_{max}]$ and $b > 0$, can be expressed as a linear combination of all powers of graph Laplacian (or adjacency) as follows, with $\alpha_{s,i} = \frac{\Phi_s^{(i)}(0)}{i!}$:

$$C'^{(s)} = \alpha_{s,0}L^0 + \alpha_{s,1}L^1 + \alpha_{s,2}L^2 + \dots.$$

- Each $C^{(s)}$ consists of different linear coefficients of power series of adjacency. Necessary masked power of adjacencies their trace and multiplications can be obtained where $M = A + I$ by:

$$C = M \odot mlp_4\left(mlp_1(C')|mlp_2(C') \odot mlp_3(C')\right),$$

- **GNNML3**'s one layer forward calculations becomes:

$$H^{(l+1)}{=}\sigma\Big(\textstyle\sum_s(C^{(s)}H^{(l)}W^{(l,s)})|mlp_5(H^{(l)}){\odot}mlp_6(H^{(l)})\Big) \qquad (3)$$

Experimental Results

- **Expressive Test** How many pairs of non-isomorphic graphs are not distinguished by the models?

Model	MLP	GCN	GAT	GIN	Chebnet	PPGN	**GNNML1**	**GNNML3**
graph8c	293K	4775	1828	386	44	**0**	333	**0**
sr25	105	105	105	105	105	105	105	105
EXP	600	600	600	600	71	**0**	600	**0**
EXP-class	50%	50%	50%	50%	82%	**100%**	50%	**100%**

- **Graphlet Counting** Can the models generalize the counting of some substructures in a given graph?

Model	MLP	GCN	GAT	GIN	Chebnet	PPGN	GNNML1	GNNML3
3-stars	1.0e-4	1.0e-4	1.0e-4	1.0e-4	1.0e-4	1.0e-4	1.0e-4	1.0e-4
triangle	3.1e-1	2.4e-1	2.5e-1	2.1e-1	2.0e-1	1.0e-4	2.4e-1	4.4e-4
tailed-tri	2.2e-1	1.4e-1	1.4e-1	1.2e-1	1.1e-1	2.6e-4	1.3e-1	3.2e-4
4-cycles	1.7e-1	1.1e-1	1.1e-1	1.2e-1	9.6e-2	3.3e-4	1.1e-1	6.6e-4

- **Spectral Ability** Can the models learn low-pass, high-pass and band-pass filtering effects?

Model	MLP	GCN	GAT	GIN	Chebnet	PPGN	GNNML1	GNNML3
Loss-pass	0.9749	0.9858	0.9811	0.9824	0.9995	0.9991	0.9994	0.9995
High-pass	0.0167	0863	0.0879	0.2934	0.9901	0.9925	0.9833	0.9909
Band-pass	0.0027	0.0051	0.0044	0.0629	0.8217	0.1041	0.3802	0.8189
Classify	50.0%	77.9%	85.3%	87.6%	98.2%	91.2%	92.8%	97.8%

Conclusion

- Except eigendecomposition in preprocessing step, GNNML3 has linear complexity.
- GNNML3 is as good as spectral graph convolution on problems depending on graph signal frequency.
- GNNML3 is as good as 3-WL equivalent GNN on problems depending on graph substructure counting and graph isomorphism test.
- GNNML3 provides trade-offs between frequency awareness and structural awareness. It would give better result on mix problems.

Acknowledgments

This work was partially supported by the Normandy Region (grant RiderNet), the French Agence National de Recherche (grant APi, ANR-18-CE23-0014) and the PAUSE Program of Collège de France.