Graph Neural Networks go grammatical

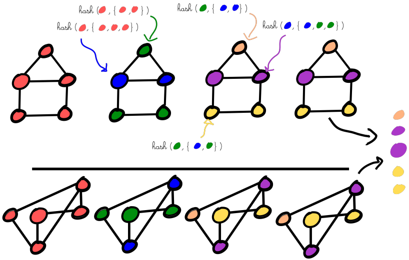
Jason Piquenot

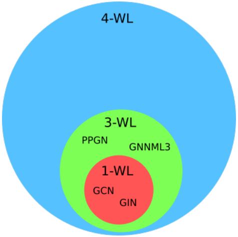
Supervisors : Sébastien Adam Romain Raveaux
Co-supervisors : Maxime Bérar Pierre Héroux Jean-Yves Ramel

statistical learning on LARge scale GRaphs (LARGR)

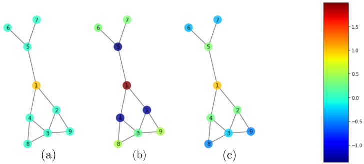
9^{th} march 2023

Weisfeiler-Lehman hierarchy [Xu et al., 2019]

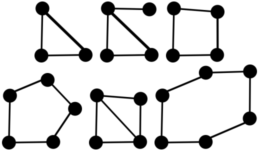




Spectral response [Balcilar et al., 2021b]



Counting power [Chen et al., 2020]



Contributions

New GNN design strategy based on Context Free Grammar (CFG).

Example of 1-WL CFG.

Grammatical Graph Neural Network (G^2N^2) a 3-WL GNN.

G^2N^2 spectral response and counting power.

In [Brijder et al., 2019] they introduced MATLANG a matrix language.

Matlang

$\mathrm{ML}(\mathcal{L})$ is a matrix language with an allowed operation set $\mathcal{L} = \{op_1, \ldots, op_n\}$, where $op_i \in \{\cdot, +, {}^{\mathbf{T}}, \mathrm{diag}, \mathrm{Tr}, 1, \odot, \times, f\}$.

Sentence

$e(X) \in \mathbb{R}$ is a sentence in $\mathrm{ML}(\mathcal{L})$ if it consists of any possible consecutive operations in $\mathcal{L}$, operating on a given matrix X and resulting in a scalar value.

As an exemple, $1^{\mathbf{T}}(X \odot \mathrm{diag}(1))\,1$ is a sentence in $\mathrm{ML}\left({}^{\mathbf{T}}, 1, \odot, \cdot, \mathrm{diag}\right)$ that computes trace of square matrix X.

$\mathrm{ML}(\mathcal{L})$-*equivalent* for matrices

Two matrices A and B in $\mathcal{M}_{m,n}(\mathbb{R})$ are said to be $\mathrm{ML}(\mathcal{L})$-*equivalent*, denoted by $A \equiv_{\mathrm{ML}(\mathcal{L})} B$, if and only if $e(A) = e(B)$ for all sentences in $\mathrm{ML}(\mathcal{L})$.

$\mathrm{ML}(\mathcal{L})$-*equivalent* for graphs

Two graphs $\mathcal{G}$ and $\mathcal{H}$ of the same order are said to be $\mathrm{ML}(\mathcal{L})$-*equivalent*, denoted by $\mathcal{G} \equiv_{\mathrm{ML}(\mathcal{L})} \mathcal{H}$, if and only if their adjacency matrices are $\mathrm{ML}(\mathcal{L})$-*equivalent*.

In [Geerts and Reutter, 2021], they proved the following theorems for the sets of operations $\mathcal{L}_1 = \{\cdot, {}^{\mathbf{T}}, 1, \text{diag}\}$ and $\mathcal{L}_3 = \{\cdot, {}^{\mathbf{T}}, 1, \text{diag}, \odot\}$.

1-WL equivalence

Two adjacency matrices are indistinguishable by the 1-WL test if and only if $e(A_G) = e(A_H)$ for all $e \in \mathcal{L}_1$. Hence, all possible sentences in $\mathcal{L}_1$ are the same for $1-$WL-*equivalent* adjacency matrices. Thus,

$$A_{\mathcal{G}} \equiv_{\text{1-WL}} A_{\mathcal{H}} \iff A_{\mathcal{G}} \equiv_{\text{ML}(\mathcal{L}_1)} A_{\mathcal{H}}$$

3-WL equivalence

Two adjacency matrices are indistinguishable by the 3-WL test if and only if $e(A_G) = e(A_H)$ for all $e \in \mathcal{L}_3$. Hence, all possible sentences in $\mathcal{L}_3$ are the same for $3-$WL-*equivalent* adjacency matrices. Thus,

$$A_{\mathcal{G}} \equiv_{\text{3-WL}} A_{\mathcal{H}} \iff A_{\mathcal{G}} \equiv_{\text{ML}(\mathcal{L}_3)} A_{\mathcal{H}}$$

For $\mathcal{L}_1 = \{\cdot, \ ^{\mathbf{T}}, 1, \text{diag}\}$, define $G_{\mathcal{L}_1} = (V, \Sigma, R, S_t)$.

$$V = \{M, V_c, V_l, S\}$$
$$\Sigma = \{A, \text{diag}, 1, \ ^{\mathbf{T}}, (,)\}$$
$$S_t = S$$

where the rules in R are

$$S \rightarrow (V_l)(V_c) \mid \text{diag}\,(S) \mid SS$$
$$M \rightarrow MM \mid (M)^{\mathbf{T}} \mid \text{diag}\,(V_c) \mid (V_c)(V_l) \mid A$$
$$V_c \rightarrow MV_c \mid (V_l)^{\mathbf{T}} \mid V_cS \mid 1$$
$$V_l \rightarrow V_lM \mid (V_c)^{\mathbf{T}} \mid SV_l$$

	Rules applied
S	
↓	$S \rightarrow V_lV_c$
V_lV_c	
↓ ↓	$V_l \rightarrow (V_c)^T, \quad V_c \rightarrow MV_c$
$(V_c)^T MV_c$	
↓ ↓ ↓	$V_c \rightarrow 1, \quad M \rightarrow MM, \quad V_c \rightarrow 1$
$(1)^T MM1$	
↓ ↓	$M \rightarrow A, \quad M \rightarrow \text{diag}(V_c)$
$(1)^T A\text{diag}(V_c)1$	
↓	$V_c \rightarrow MV_c$
$(1)^T A\text{diag}(MV_c)1$	
↓ ↓	$M \rightarrow A, \quad V_c \rightarrow 1$
$(1)^T A\text{diag}(A1)1$	

Theorem (ML $(\mathcal{L}_1)$ Reduced CFG)

The following CFG denoted by r-$G_{\mathcal{L}_1}$ is as expressive as 1-WL.

$$V_c \to \mathrm{diag}\,(V_c)\,V_c \mid AV_c \mid 1$$

Corollary (GNNML1 CFG)

The following CFG, as expressive than ML $(\mathcal{L}_1)$ *represents GNNML1 [Balcilar et al., 2021a].*

$$V_c \to V_c \odot V_c \mid AV_c \mid 1$$

GNNML1 node update:

$$\begin{aligned} H^{(l+1)} = \sigma(H^{(l)} \cdot W^{(l,1)} + A \cdot H^{(l)} \cdot W^{(l,2)} \\ + H^{(l)} \cdot W^{(l,3)} \odot H^{(l)} \cdot W^{(l,4)}) \end{aligned}$$

Proposition (GCN CFG)

The following CFG is strictly less expressive than $\mathrm{ML}(\mathcal{L}_1)$.

$$V_c \to C_1 V_c \mid \cdots \mid C_k V_c \mid 1$$

When C_s *is include in* $\mathrm{ML}(\mathcal{L}_1)$.

As an example, the following CFG, strictly less expressive than $\mathrm{ML}(\mathcal{L}_1)$ represents GCN [Kipf and Welling, 2017].

$$V_c \to C V_c \mid 1$$

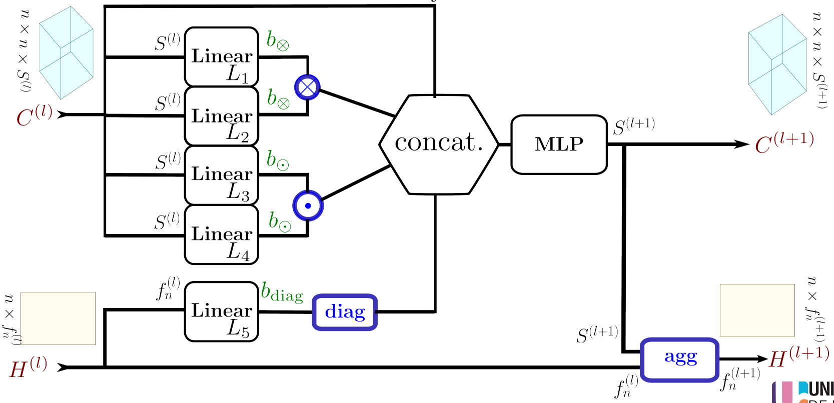
Where $C = \mathrm{diag}\left((A + \mathrm{I})1\right)^{-\frac{1}{2}} (A + \mathrm{I})\mathrm{diag}\left((A + \mathrm{I})1\right)^{-\frac{1}{2}}$

Theorem (ML $(\mathcal{L}_3)$ Reduced CFG)

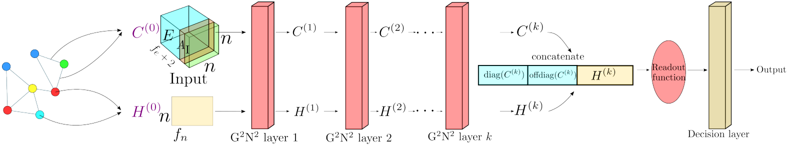
The following CFG denoted by r-$G_{\mathcal{L}_3}$ is as expressive as 3-WL.

$$V_c \to MV_c \mid 1$$
$$M \to (M \odot M) \mid MM \mid \text{diag}(V_c) \mid A$$

G^2N^2 layer :



G^{2}N^2 architecture :



G^{2}N^2 update equation

$$\mathcal{C}^{(l+1)} = mlp(\mathcal{C}^{(l)}|L_1(\mathcal{C}^{(l)})\odot L_2(\mathcal{C}^{(l)})|$$
$$L_3(\mathcal{C}^{(l)})\cdot L_4(\mathcal{C}^{(l)})|\text{diag}(L_5(H^{(l)}))),$$

$$H^{(l+1)} = \sum_{i=1}^{S^{(l+1)}} \mathcal{C}_i^{(l+1)} H^{(l)} W^{(l,i)}.$$

Theorem (G^2N^2 in the WL hierarchy)

G^2N^2 is as expressive as 3-WL.

Table: The accuracy on EXP and SR25 datasets denotes the ratio of pairs of non isomorphic respectively 1-WL equivalent and 3-WL equivalent graphs that are separate by the model.

Method	EXP	SR25
1-WL-bounded GNN	0%	0%
CHEBNET	87%	0%
3-WL-bounded GNN	**100%**	0%
PPGN	**100%**	0%
G^2N^2	**100%**	0%
I^2-GNN	**100%**	**100%**

Theorem (G^2N^2 counting power)

G^2N^2 can count chordal cycle and cycle up to lenght 6 at edge level.

Table: G^2N^2 normalised MAE on counting substructures at edge level.

triangle	4-cycle	5-cycle	6-cycle	chordal cycle
3.99e-04	4.55e-04	2.93e-03	3.58e-03	1.56e-04

Theorem (G^2N^2 spectral response)

G^2N^2 can approximate low-pass, high-pass and band-pass filter in the spectral domain.

Table: R^2 score on spectral filtering node regression problems. Results are median of 10 different runs.

Method	Low-pass	High-pass	Band-pass
MLP	0.9749	0.0167	0.0027
GCN	0.9858	0.0863	0.0051
GAT	0.9811	0.0879	0.0044
GIN	0.9824	0.2934	0.0629
CHEBNET	**0.9995**	0.9901	**0.8217**
PPGN	0.9991	**0.9925**	0.1041
GNNML1	0.9994	0.9833	0.3802
GNNML3	**0.9995**	**0.9909**	**0.8189**
G^2N^2	**0.9996**	**0.9994**	**0.8206**

On the left G^2N^2's prediction, on the center the input and on the right, the target

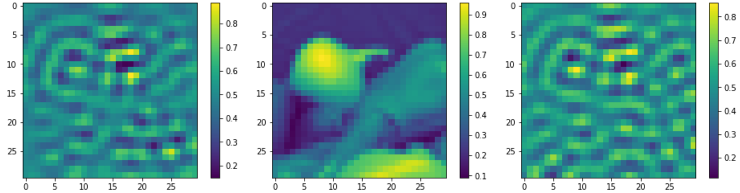


Table: Results on QM9 dataset focusing on the best methods. The metric is MAE, the lower, the better.

Target	I^2-GNN	PPGN(1)	PPGN(12sm)	G^2N^2(12)
μ	0.428	0.0934	0.485	**0.0621**
α	0.230	0.318	0.504	**0.119**
ϵ_{homo}	0.00261	**0.00174**	0.126	0.0312
ϵ_{lumo}	0.00267	**0.0021**	0.138	0.0333
$\Delta\epsilon$	0.0038	**0.0029**	0.182	0.0427
R^2	18.64	3.78	17.17	**0.992**
ZPVE	**0.00014**	0.000399	0.0105	0.00426
U_0	0.211	**0.022**	14.41	1.71
U	0.206	**0.0504**	14.41	1.72
H	0.269	**0.0294**	14.41	1.71
G	0.261	**0.024**	14.41	1.72
C_v	0.0730	0.144	0.192	**0.0581**

Table: Results of G^2N^2 on TUD dataset compared to the best competitor. The metric is accuracy, the higher, the better.

Dataset	G^2N^2	rank	Best GNN competitor
MUTAG	92.0±4.3	2	92.2±7.5
PTC	71.8±6.7	1	68.2±7.2
Proteins	77.8±3.2	1	77.4±4.9
NCI1	80.2±2.1	8	83.5±2.0
IMDB-B	76.8±2.8	2	77.8±3.3
IMDB-M	54.0±2.93	2	54.3±3.3

G^2N^2 assets

3-WL equivalent
Large spectral response
Counting power at **edge level**
Direct from CFG

G^2N^2 drawbacks

Time complexity $O\left(n^3\right)$
Memory complexity $O\left(n^2\right)$

Balcilar, M., Héroux, P., Gauzere, B., Vasseur, P., Adam, S., and Honeine, P. (2021a).
Breaking the limits of message passing graph neural networks.
In *International Conference on Machine Learning*, pages 599–608. PMLR.

Balcilar, M., Renton, G., Héroux, P., Gaüzère, B., Adam, S., and Honeine, P. (2021b).
Analyzing the expressive power of graph neural networks in a spectral perspective.
In *International Conference on Learning Representations*.

Brijder, R., Geerts, F., den Bussche, J. V., and Weerwag, T. (2019).
On the expressive power of query languages for matrices.
ACM Trans. Database Syst., 44(4):15:1–15:31.

Chen, Z., Chen, L., Villar, S., and Bruna, J. (2020).
Can graph neural networks count substructures?
Advances in neural information processing systems, 33:10383–10395.

Geerts, F. and Reutter, J. L. (2021).
Expressiveness and approximation properties of graph neural networks.
In *International Conference on Learning Representations*.

Kipf, T. N. and Welling, M. (2017).
Semi-supervised classification with graph convolutional networks.
In *5th International Conference on Learning Representations*.

Xu, K., Hu, W., Leskovec, J., and Jegelka, S. (2019).
How powerful are graph neural networks?
In *International Conference on Learning Representations*.

Thank you for your attention.

Our paper is available here.

Graph Neural Network go Grammatical

https://arxiv.org/abs/2303.01590