Group Meeting - April 24, 2020

Paper review & Research progress

Truong Son Hy *

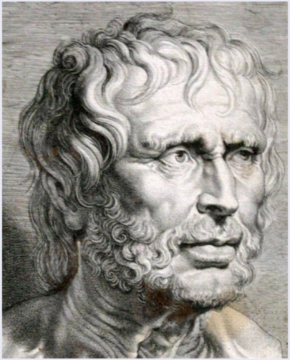
*Department of Computer Science
The University of Chicago

Ryerson Physical Lab

Stocism (1)

Seneca the Younger

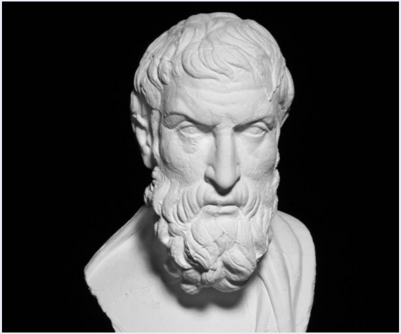
The greatest obstacle to living is expectancy, which hangs upon tomorrow and loses today. **The whole future lies in uncertainty: live immediately.**



Stocism (2)

Epictetus

Just keep in mind: the more we value things outside our control, the less control we have.



Plato's Philosopher King

Plato

1. There will be no end to the troubles of states, or of humanity itself, till **philosophers become kings** in this world, or till those we now call kings and rulers really and truly become philosophers, and political power and philosophy thus come into the same hands.
2. Those who are too smart to engage in politics are punished by being governed by those who are dumber.

Note: The second way never happened!

Papers

1. **Provably Powerful Graph Networks**:

   `https://papers.nips.cc/paper/8488-provably-powerful-graph-networks.pdf`

2. **On the equivalence between graph isomorphism testing and function approximation with GNNs**:

   `https://arxiv.org/pdf/1905.12560.pdf`

Provably Powerful Graph Networks (1)

Measurement of expressive power of GNNs

- The Weisfeiler-Lehman (WL) graph isomorphism test was used to measure the expressive power of graph neural networks (GNNs).
- **Morris et al., 2018** & **Xu et al., 2019**: Popular message passing GNN cannot distinguish between graphs that are indistinguishable by the 1-WL test.

Provably Powerful Graph Networks (1)

Measurement of expressive power of GNNs

- The Weisfeiler-Lehman (WL) graph isomorphism test was used to measure the expressive power of graph neural networks (GNNs).
- **Morris et al., 2018** & **Xu et al., 2019**: Popular message passing GNN cannot distinguish between graphs that are indistinguishable by the 1-WL test.

Main results

1. *k*-order networks can distinguish between non-isomorphic graphs as good as the *k*-WL tests, which are provably stronger than the 1-WL test for $k > 2$. **Space/time computational cost**: out of PSPACE.
2. Building a provably stronger, simple and scalable model: a reduced 2-order network that has a provable 3-WL expressive model.

Provably Powerful Graph Networks (2)

Set: $\{a, b, .., c\}$. Ordered set (tuple): $(a, b, .., c)$. Multi set: $\{\{a, b, .., c\}\}$.
$g \in S_n$ acts on multi-indices $i \in [n]^k$ entrywise by:

$$g(i) = (g(i_1), g(i_2), .., g(i_k))$$

S_n acts on k-tensors $X \in \mathbb{R}^{n^k \times a}$ by:

$$(g \cdot X)_{i,j} = X_{g^{-1}(i),j}$$

where $i \in [n]^k$ and $j \in [a]$.

Provably Powerful Graph Networks (3)

Construct networks by concatenating maximally expressive linear equivariant layers:

$$F = m \circ h \circ L_d \circ \sigma \circ \cdots \circ \sigma \circ L_1$$

where:

- $L_i : \mathbb{R}^{n^{k_i} \times a_i} \to \mathbb{R}^{n^{k_{i+1}} \times a_{i+1}}$ are equivariant linear layers, and $k = \max_{i \in [d+1]} k_i$.

$$L_i(g \cdot X) = g \cdot L_i(X) \qquad \forall g \in S_n \qquad \forall X \in \mathbb{R}^{n^{k_i} \times a_i}$$

- σ is an entrywise non-linear activation.

$$\sigma(X)_{i,j} = \sigma(X_{i,j})$$

- $h : \mathbb{R}^{n^{k_{d+1}} \times a_{d+1}} \to \mathbb{R}^{a_{d+2}}$ is an invariant linear layer.

$$h(g \cdot X) = h(X) \qquad \forall g \in S_n \qquad \forall X \in \mathbb{R}^{n^{k_{d+1}} \times a_{d+1}}$$

- m is a MLP.

Provably Powerful Graph Networks (4)

Let $G = (V, E, d)$ be a colored graph where $|V| = n$ and $d : V \to \Sigma$ defines the color attached to each vertex in V, Σ is a set of colors. Two graphs G and G' are called **isomorphic** if there exists an **edge and color preserving** bijection $\phi : V \to V'$.

The k-WL test $(k = 1, 2, .., n)$ constructs a coloring of k-tuples of vertices, that is $c : V^k \to \Sigma$. Testing isomorphism of two graphs G and G' are performed by comparing the histograms of colors produced by the k-WL algorithms.

Provably Powerful Graph Networks (5)

Coloring of k-tuples using a tensor $C \in \Sigma^{n^k}$, where $C_i \in \Sigma$, $i \in [n]^k$ denotes the color of the k-tuple $v_i = (v_{i_1}, .., v_{i_k}) \in V^k$.

Two k-tuples i and i' have the same **isomorphism type** (i.e., get the same color, $C_i = C_{i'}$) if for all $q, r \in [k]$:

- $v_{i_q} = v_{i_r} \Leftrightarrow v_{i'_q} = v_{i'_r}$
- $d(v_{i_q}) = d(v_{i'_q})$
- $(v_{i_r}, v_{i_q}) \in E \Leftrightarrow (v_{i'_r}, v_{i'_q}) \in E$

Clearly, if G and G' are two isomorphic graphs then there exists $g \in S_n$ so that:

$$g \cdot C'^0 = C^0$$

Provably Powerful Graph Networks (6)

The algorithms refine the colorings $C^l, l = 1, 2, ..$ until the coloring does not change further, that is, the subsets of k-tuples with same colors do not get further split to different color groups.

(Douglas, 2011)

No more than $l = poly(n)$ iterations are required.

Neighborhoods of a k-tuple $i \in [n]^k$:

$$N_j(i) = \left\{ (i_1, .., i_{j-1}, i', i_{j+1}, .., i_k) \middle| i' \in [n] \right\}$$

The coloring update rules are:

$$C_i^l = \left(C_i^{l-1}, \left(\{\{ C_j^{l-1} | j \in N_v(i) \}\} \middle| v \in [k] \right) \right)$$

Provably Powerful Graph Networks (7)

Theorem 1

Given two graphs $G = (V, E, d)$ and $G' = (V', E', d')$ that can be distinguished by the k-WL graph isomorphism test, there exists a k-order network F so that $F(G) \neq F(G')$. On the other direction, for every two isomorphic graphs G and G', and k-order network F so that $F(G) = F(G')$.

Sketch of the proof:

1. It is **trivial** to construct a k-network that returns the same output for every pair of isomorphic graphs G and G'.
2. For non-isomorphic graphs, it is more challenging:
   - Encode a multiset X using a set of S_n-invariant functions called the **Power-sum Multi-symmetric Polynomials** (PMP):

$$p_\alpha(X) = \sum_{i=1}^{n} x_i^\alpha, \qquad X \in \mathbb{R}^{n \times a}$$

where $y^\alpha = y_1^{\alpha_1} \cdot y_2^{\alpha_2} \cdots y_a^{\alpha_a}$ for $\alpha = (\alpha_1, ..., \alpha_a) \in [n]^a$ is a multi-index, and $|\alpha| = \sum_{j=1}^{a} \alpha_j$.

Sketch of the proof:

1. It is **trivial** to construct a k-network that returns the same output for every pair of isomorphic graphs G and G'.
2. For non-isomorphic graphs, it is more challenging:
   - Encode a multiset X using a set of S_n-invariant functions called the **Power-sum Multi-symmetric Polynomials** (PMP).
   - $u(X) = \left(p_\alpha(X) \middle| |\alpha| \leq n \right)$ is a unique representation of the multi-set $X \in \mathbb{R}^{n \times a}$.
   - Apply the Universal approximation theorem (Cybenko, 1989; Hornik, 1991) that any polynomical function can be approximated/replaced by an MLP.

Provably Powerful Graph Networks (10)

Table 1: Graph Classification Results on the datasets from Yanardag and Vishwanathan (2015)

dataset	MUTAG	PTC	PROTEINS	NCI1	NCI109	COLLAB	IMDB-B	IMDB-M
size	188	344	1113	4110	4127	5000	1000	1500
classes	2	2	2	2	2	3	2	3
avg node #	17.9	25.5	39.1	29.8	29.6	74.4	19.7	13
Results								
GK (Shervashidze et al., 2009)	81.39±1.7	55.65±0.5	71.39±0.3	62.49±0.3	62.35±0.3	NA	NA	NA
RW (Vishwanathan et al., 2010)	79.17±2.1	55.91±0.3	59.57±0.1	> 3 days	NA	NA	NA	NA
PK (Neumann et al., 2016)	76±2.7	59.5±2.4	73.68±0.7	82.54±0.5	NA	NA	NA	NA
WL (Shervashidze et al., 2011)	84.11±1.9	57.97±2.5	**74.68±0.5**	**84.46±0.5**	**85.12±0.3**	NA	NA	NA
FGSD (Verma and Zhang, 2017)	**92.12**	**62.80**	73.42	79.80	78.84	**80.02**	73.62	**52.41**
AWE-DD (Ivanov and Burnaev, 2018)	NA	NA	NA	NA	NA	73.93±1.9	**74.45 ± 5.8**	51.54 ±3.6
AWE-FB (Ivanov and Burnaev, 2018)	87.87±9.7	NA	NA	NA	NA	70.99 ± 1.4	73.13 ±3.2	51.58 ± 4.6
DGCNN (Zhang et al., 2018)	85.83±1.7	58.59±2.5	75.54±0.9	74.44±0.5	NA	73.76±0.5	70.03±0.9	47.83±0.9
PSCN (Niepert et al., 2016)(k=10)	88.95±4.4	62.29±5.7	75±2.5	76.34±1.7	NA	72.6±2.2	71±2.3	45.23±2.8
DCNN (Atwood and Towsley, 2016)	NA	NA	61.29±1.6	56.61± 1.0	NA	52.11±0.7	49.06±1.4	33.49±1.4
ECC (Simonovsky and Komodakis, 2017)	76.11	NA	NA	76.82	75.03	NA	NA	NA
DGK (Yanardag and Vishwanathan, 2015)	87.44±2.7	60.08±2.6	75.68±0.5	80.31±0.5	80.32±0.3	73.09±0.3	66.96±0.6	44.55±0.5
DiffPool (Ying et al., 2018)	NA	NA	**78.1**	NA	NA	75.5	NA	NA
CCN (Kondor et al., 2018)	**91.64±7.2**	**70.62±7.0**	NA	76.27±4.1	75.54±3.4	NA	NA	NA
Invariant Graph Networks (Maron et al., 2019a)	83.89±12.95	58.53±6.86	76.58±5.49	74.33±2.71	72.82±1.45	78.36±2.47	72.0±5.54	48.73±3.41
GIN (Xu et al., 2019)	89.4±5.6	64.6±7.0	76.2±2.8	82.7±1.7	NA	80.2±1.9	**75.1±5.1**	**52.3±2.8**
1-2-3 GNN (Morris et al., 2018)	86.1±	60.9±	75.5±	76.2±	NA	NA	74.2±	49.5±
Ours 1	90.55±8.7	66.17±6.54	77.2±4.73	**83.19±1.11**	**81.84±1.85**	80.16±1.11	72.6±4.9	50±3.15
Ours 2	88.88±7.4	64.7±7.46	76.39±5.03	81.21±2.14	**81.77±1.26**	**81.38±1.42**	72.2±4.26	44.73±7.89
Ours 3	89.44±8.05	62.94±6.96	76.66±5.59	80.97±1.91	**82.23±1.42**	**80.68±1.71**	73±5.77	50.46±3.59
Rank	3^{rd}	2^{nd}	2^{nd}	2^{nd}	2^{nd}	1^{st}	6^{th}	5^{th}

Table 2: Regression, the QM9 dataset.

Target	DTNN	MPNN	123-gnn	Ours 1	Ours 2
μ	0.244	0.358	0.476	**0.231**	**0.0934**
α	0.95	0.89	**0.27**	0.382	0.318
ϵ_{homo}	0.00388	0.00541	0.00337	**0.00276**	**0.00174**
ϵ_{lumo}	0.00512	0.00623	0.00351	**0.00287**	**0.0021**
Δ_ϵ	0.0112	0.0066	0.0048	**0.00406**	**0.0029**
$\langle R^2 \rangle$	17	28.5	22.9	**16.07**	**3.78**
$ZPVE$	0.00172	0.00216	**0.00019**	0.00064	0.000399
U_0	-	-	0.0427	0.234	**0.022**
U	-	-	0.111	0.234	**0.0504**
H	-	-	0.0419	0.229	**0.0294**
G	-	-	0.0469	0.238	**0.024**
C_v	0.27	0.42	**0.0944**	0.184	0.144

On the equivalence between graph isomorphism testing and function approximation with GNNs (1)

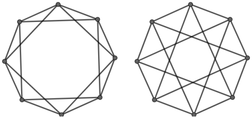


Figure 2: The Circular Skip Link graphs $G_{n,k}$ are undirected graphs in n nodes $q_0, \ldots, q_{n-1}$ so that $(i, j) \in E$ if and only if $|i - j| \equiv 1$ or $k \pmod n$. In this figure we depict (left) $G_{8,2}$ and (right) $G_{8,3}$. It is very easy to check that $G_{n,k}$ and $G_{n',k'}$ are not isomorphic unless $n = n'$ and $k \equiv \pm k' \pmod n$. Both 1-WL and G-invariant networks fail to distinguish them.

Theorem

Order-2 Graph G-invariant Networks cannot distinguish between non-isomorphic regular graphs with the same degree.

On the equivalence between graph isomorphism testing and function approximation with GNNs (2)

Definition 5 (Ring-GNN). *Given a graph in n nodes with both node and edge features in $\mathbb{R}^d$, we represent it with a matrix $A \in \mathbb{R}^{n\times n\times d}$. [17] shows that all linear equivariant layers from $\mathbb{R}^{n\times n}$ to $\mathbb{R}^{n\times n}$ can be expressed as $L_\theta(A) = \sum_{i=1}^{15} \theta_i L_i(A) + \sum_{i=16}^{17} \theta_i \overline{L}_i$, where the $\{L_i\}_{i=1,\ldots,15}$ are the 15 basis functions of all linear equivariant functions from $\mathbb{R}^{n\times n}$ to $\mathbb{R}^{n\times n}$, $\overline{L}_{16}$ and $\overline{L}_{17}$ are the basis for the bias terms, and $\theta \in \mathbb{R}^{17}$ are the parameters that determine L. Generalizing to an equivariant linear layer from $\mathbb{R}^{n\times n\times d}$ to $\mathbb{R}^{n\times n\times d'}$, we set $L_\theta(A)_{\cdot,\cdot,k'} = \sum_{k=1}^{d}\sum_{i=1}^{15} \theta_{k,k',i} L_i(A_{\cdot,\cdot,i}) + \sum_{i=16}^{17} \theta_{k,k',i}\overline{L}_i$, with $\theta \in \mathbb{R}^{d\times d'\times 17}$.*

With this formulation, we now define a Ring-GNN with T layers. First, set $A^{(0)} = A$. In the t^{th} layer, let

$$
\begin{aligned}
B_1^{(t)} &= \rho(L_{\alpha^{(t)}}(A^{(t)})) \\
B_2^{(t)} &= \rho(L_{\beta^{(t)}}(A^{(t)}) \cdot L_{\gamma^{(t)}}(A^{(t)})) \\
A^{(t+1)} &= k_1^{(t)} B_1^{(t)} + k_2^{(t)} B_2^{(t)}
\end{aligned}
$$

where $k_1^{(t)}, k_2^{(t)} \in \mathbb{R}$, $\alpha^{(t)}, \beta^{(t)}, \gamma^{(t)} \in \mathbb{R}^{d^{(t)}\times d'^{(t)}\times 17}$ are learnable parameters. If a scalar output is desired, then in the general form, we set the output to be $\theta_S \sum_{i,j} A_{ij}^{(T)} + \theta_D \sum_{i,i} A_{ii}^{(T)} + \sum_i \theta_i \lambda_i(A^{(T)})$, where $\theta_S, \theta_D, \theta_1, \ldots, \theta_n \in \mathbb{R}$ are trainable parameters, and $\lambda_i(A^{(T)})$ is the i-th eigenvalue of $A^{(L)}$.

On the equivalence between graph isomorphism testing and function approximation with GNNs (3)

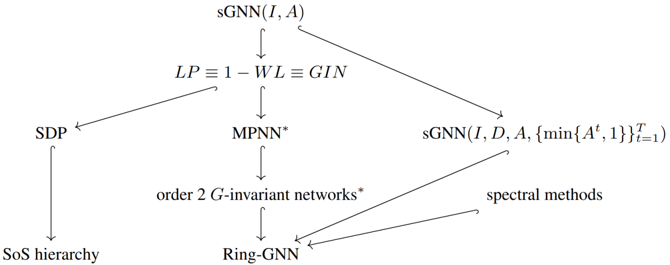


Figure 1: Relative comparison of function classes in terms of their ability to solve graph isomorphism. *Note that, on one hand GIN is defined by [30] as a form of message passing neural network justifying the inclusion GIN $\hookrightarrow$ MPNN. On the other hand [17] shows that message passing neural networks can be expressed as a modified form of order 2 G-invariant networks (which may not coincide with the definition we consider in this paper). Therefore the inclusion GIN $\hookrightarrow$ order 2 G-invariant networks has yet to be established rigorously.