Relational Pooling for Graph Representations

Ryan L. Murphy [1]

(with Balasubramaniam Srinivasan[2], Vinayak Rao[1], Bruno Ribeiro[2])

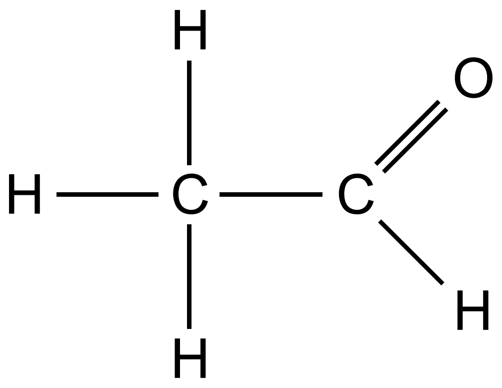
[1]Department of Statistics
[2]Department of Computer Science
Purdue University, West Lafayette, IN, USA

PURDUE
UNIVERSITY

ArXiv

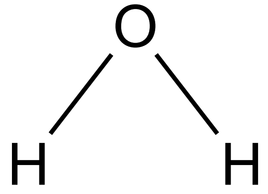
Learning Graph Representations

- A graph representation function f maps graphs to real-valued vectors
  - Graphs can have vertex/edge features
- Example: representations for end-to-end supervised learning on graphs



$\xrightarrow{f} \boldsymbol{h} \in \mathbb{R}^d$

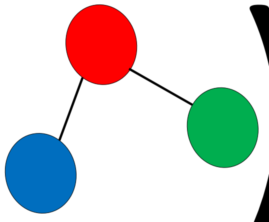
Use $\boldsymbol{h}$ to predict properties of the molecules



$\xrightarrow{f} \boldsymbol{h} \in \mathbb{R}^d$

Permutation-Invariance of Learned Representations

- An adjacency matrix $\boldsymbol{A}$ in the data is not the only valid such matrix, any permuted version, denoted $\boldsymbol{A}^{(\pi)}$, is also valid

$f($  $)$

$$= f\left(\begin{matrix} & 1 & 2 & 3 \\ 1 & 0 & 1 & 0 \\ 2 & 1 & 0 & 1 \\ 3 & 0 & 1 & 0 \end{matrix}\right) = f\left(\begin{matrix} & 1 & 2 & 3 \\ 1 & 0 & 1 & 1 \\ 2 & 1 & 0 & 0 \\ 3 & 1 & 0 & 0 \end{matrix}\right)$$

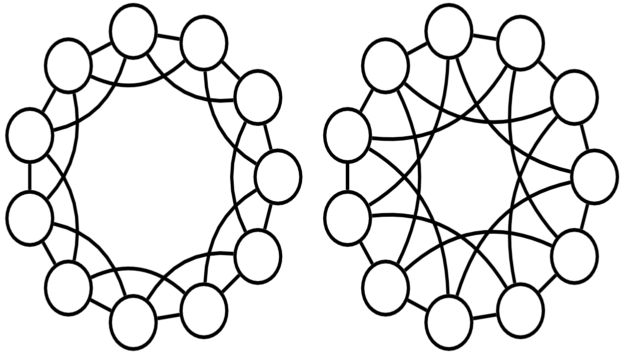
$A^{(123)}$ $A^{(213)}$

Current Representations are Limited

- Example: For GNNs, a current state-of-the-art for learning permutation-invariant representations, we have:

> **Theorem:(Xu et al. 2019, Morris et al. 2019):** WL[1] GNNs are **no more powerful than the Weisfeiler-Lehman (WL) algorithm** for graph isomorphism testing.

- **WL[1] GNNs can't perform CSL task:**
  - Cycle graphs with skip links of length R
  - Task: given graph, predict R
  - WL[1] GNNs fails



- **Relational Pooling** will help overcome such limitations

Relational Pooling

- Given graph $G = (\boldsymbol{A}, \boldsymbol{X})$ with n vertices, where rows of $\boldsymbol{X}$ are node attributes

$$\bar{\bar{f}}(\boldsymbol{A}, \boldsymbol{X}) = \frac{1}{n!} \sum_{\pi} \vec{f}(\boldsymbol{A}^{(\pi)}, \boldsymbol{X}^{(\pi)})$$

Any permutation-sensitive graph function

Theorem 2. 1: RP is **universal graph representation** if $\vec{f}$ is expressive enough.

- RP is a most-powerful representation
  - but intractable, must be approximated

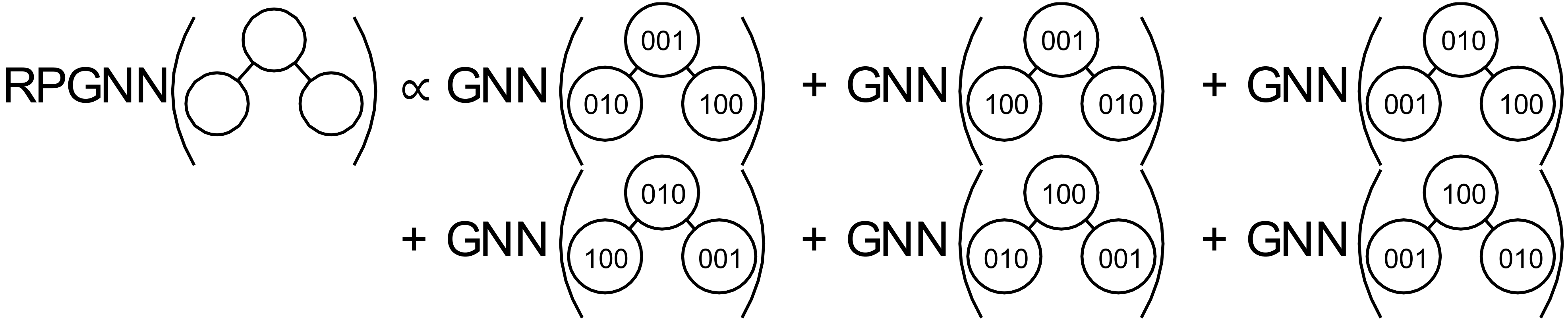
A Case-Study: Making GNNs more expressive

- Define a **permutation-sensitive GNN**

(1) add unique IDs as node features

(2) run any GNN

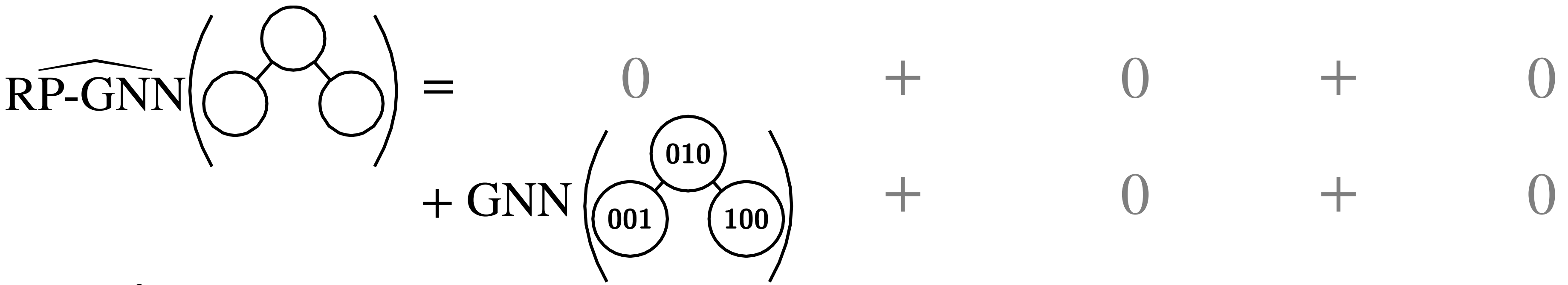
RP-GNN: sum over all permutations of IDs



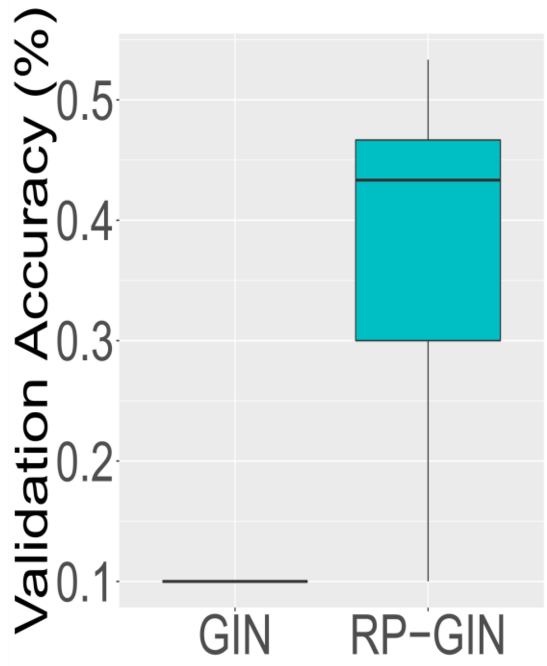
Theorem 2. 2: RP-GNN is more powerful than state-of-the-art GNNs

One *tractability approach*: stochastic optimization (π-SGD)

- At each epoch, just sample one set of permutation-sensitive IDs



- <u>CSL</u> task w/ 10 classes (graphs with 41 vertices), **RP-GNN*** to predict the class
- We also observed promising results wrapping RP around GNNs for molecules
- **Take home:** adding stochastic positional IDs is a simple way to make GNNs more powerful!



*state-of-the-art Graph Isomorphism Network of Xu et. al. 2019

Approximate Permutation-Invariance

- Estimating most-expressive RP with tractability strategies is only approximately permutation-invariant
- But learning more expressive models approximately opens up interesting new research directions

Summary

- RP provides most-expressive representations, learned approximately
  - Promising new research direction
- Our poster includes details on
  - **more tractability strategies**
  - choices for $\vec{f}$, like **CNNs and RNNs**, now valid under RP

Poster: ***Relational Pooling for Graph Representations,***
Today 06:30 -- 09:00 PM Pacific Ballroom #174

ArXiv →