Graph Neural Networks: introduction

Piotr Gaiński

Jagiellonian University

Table of Content

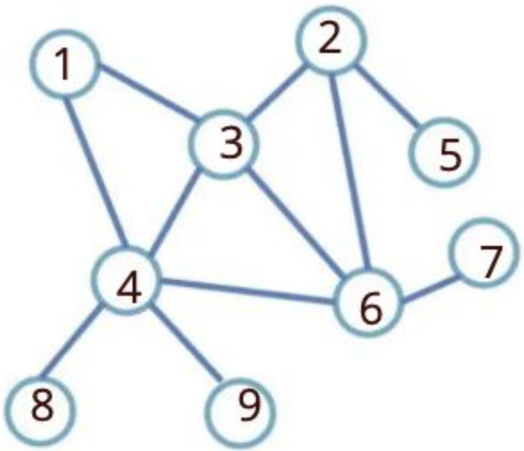
What are graphs?

Definition of a graph

$$G = (V, E)$$

$$V = \{v_i : i \in \{1,2,\ldots,N\}\}$$

$$E \subseteq \{(v_i, v_j) : v_i, v_j \in V\}$$



Many Types of Graphs

Image credit: Medium

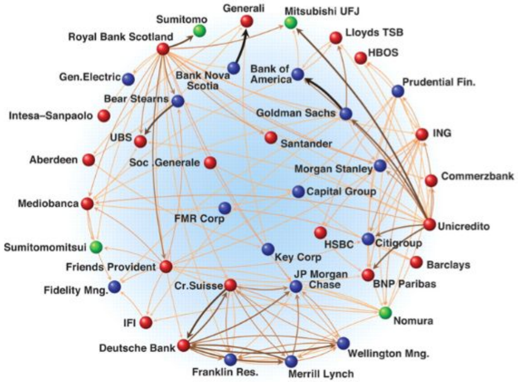


Image credit: Science

Image credit: Lumen Learning

Many Types of Graphs

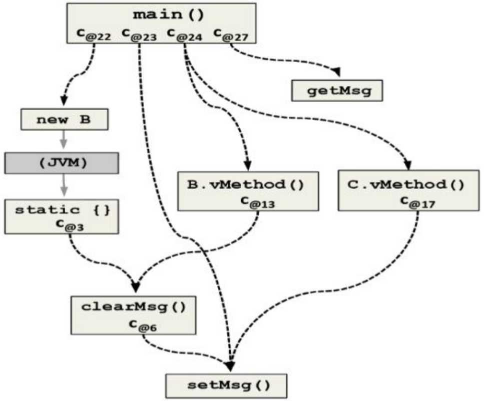


Image credit: ResearchGate

Code Graphs

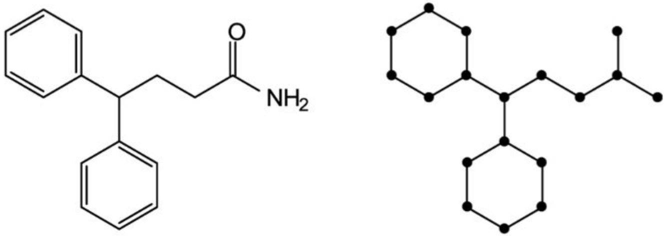


Image credit: MDPI

Molecules

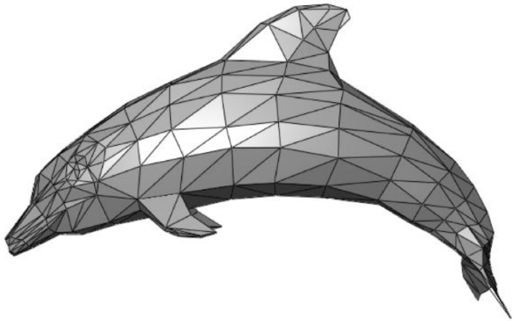
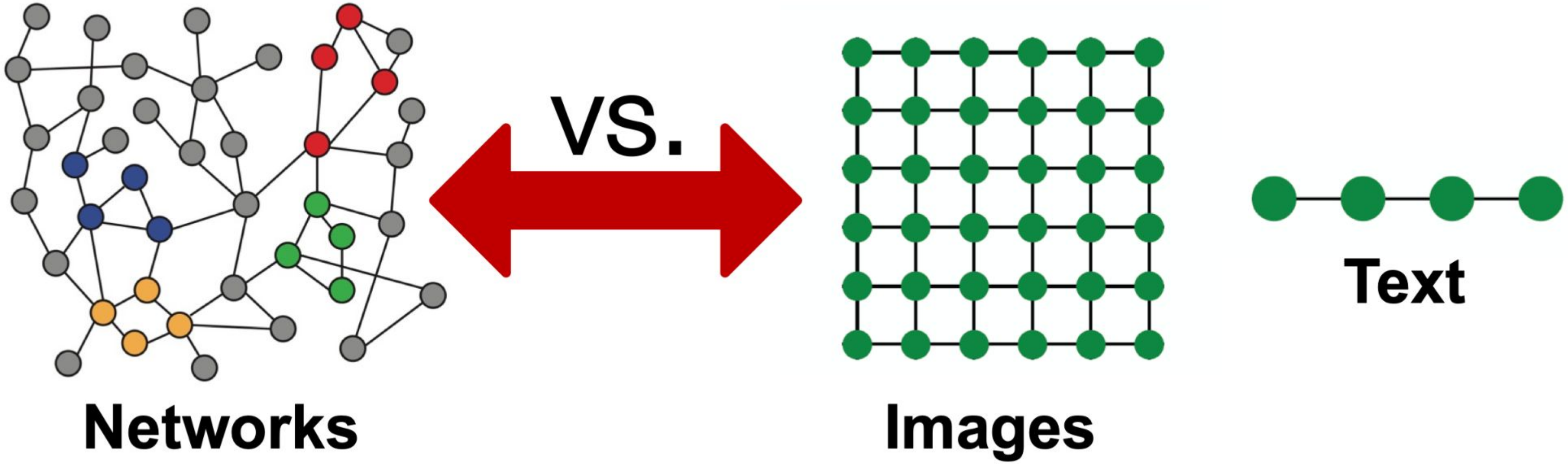


Image credit: Wikipedia

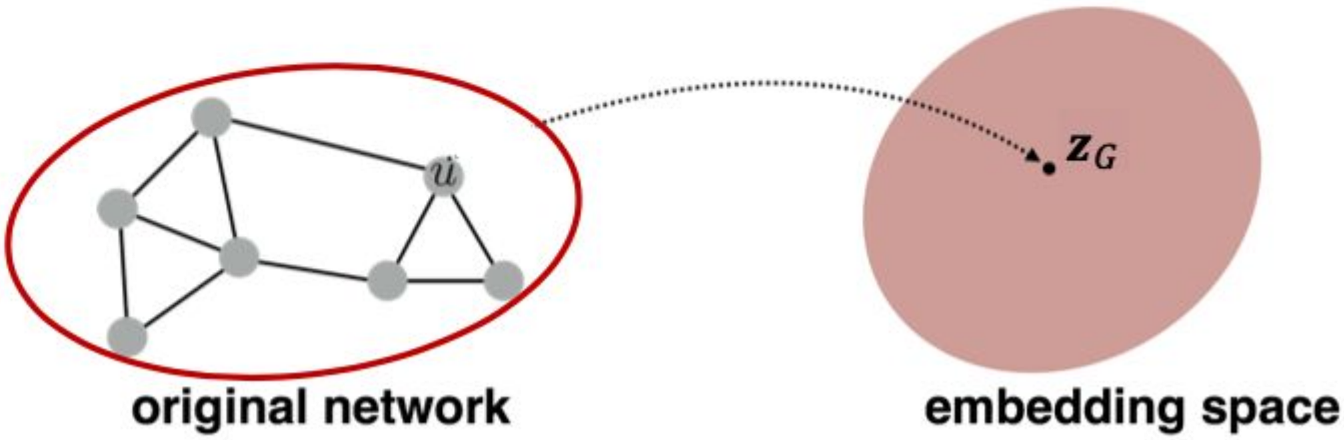
3D Shapes

Graphs are complex!

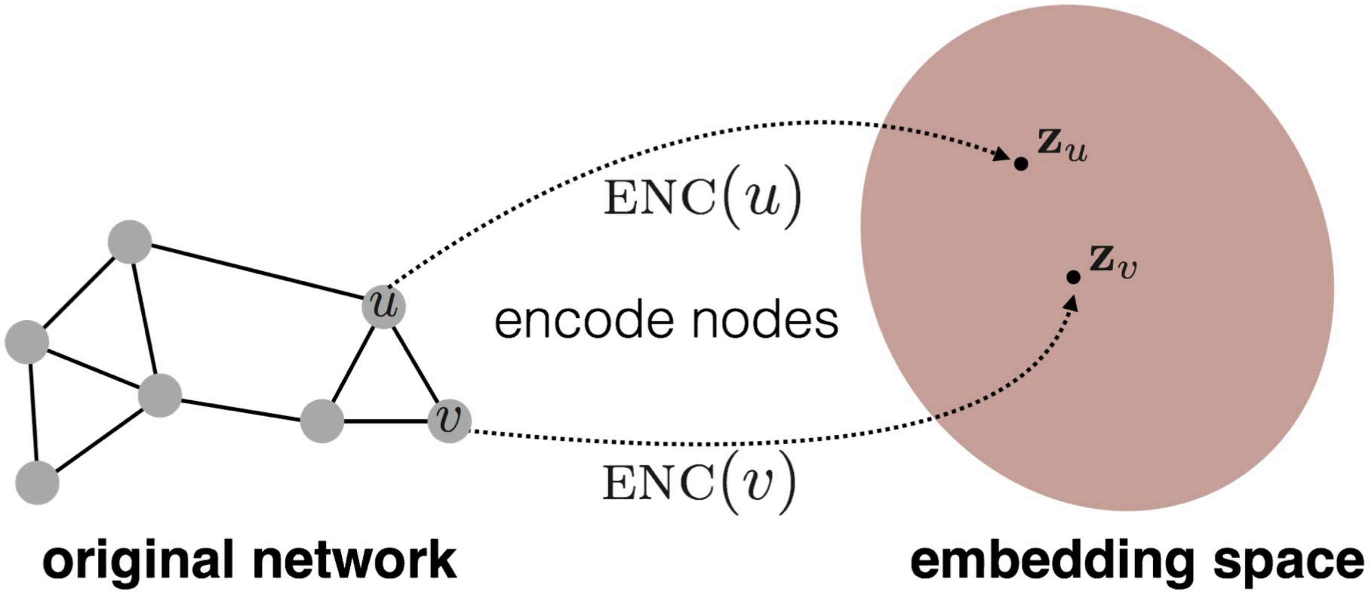


How to embed a graph?

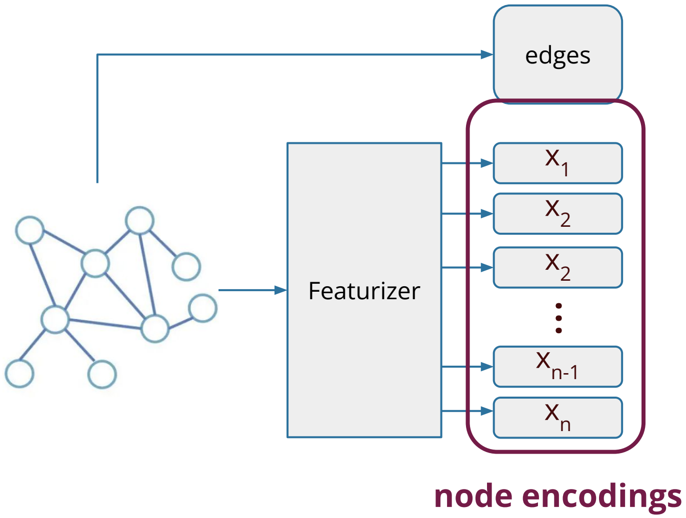
Embedding Space



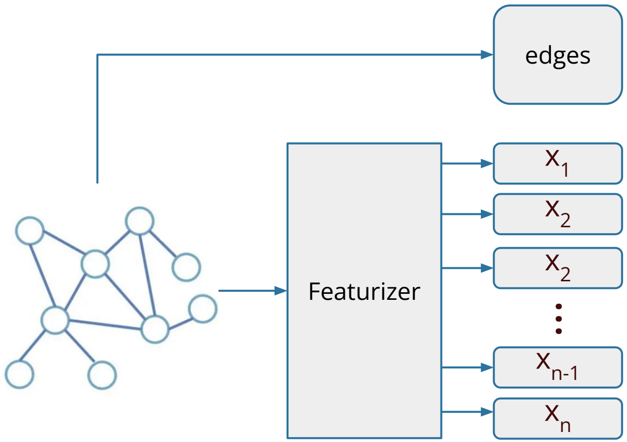
Embedding Space

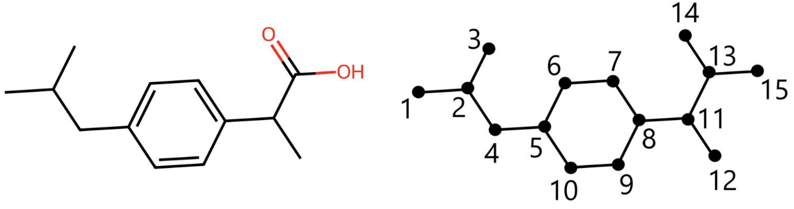


Node Encodings



Node Encodings

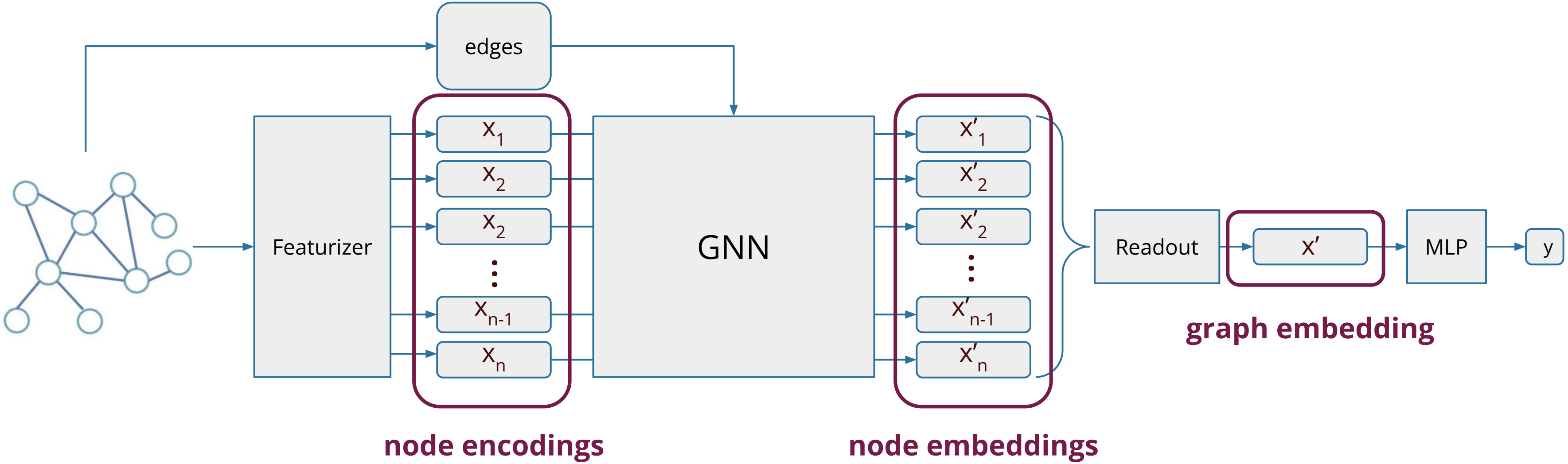




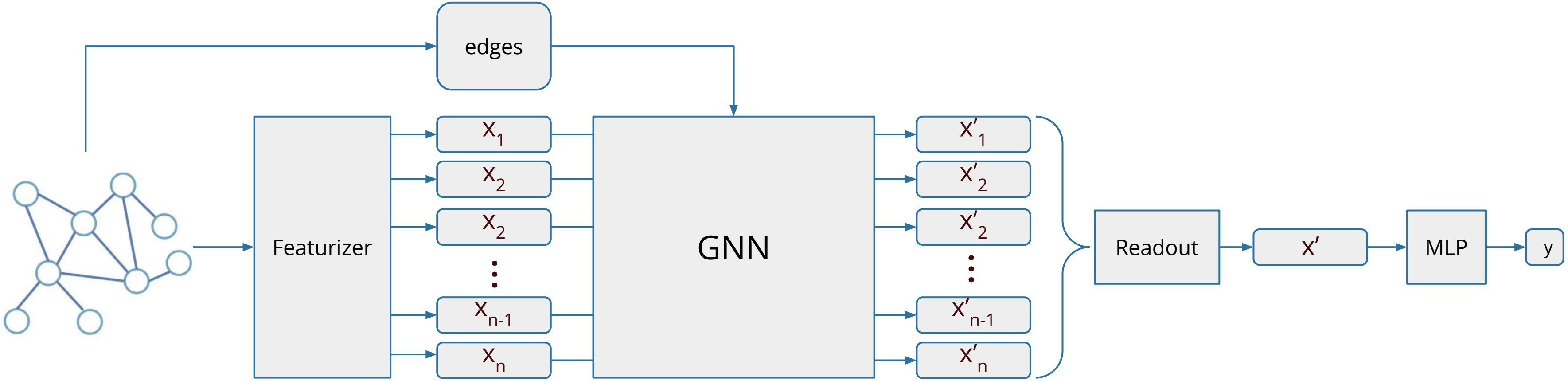
$$X = \begin{array}{c} \\ 1 \\ 2 \\ 3 \\ 4 \\ \vdots \\ 13 \\ 14 \\ 15 \end{array} \begin{array}{c} \begin{array}{cccccc} \mathrm{C} & \mathrm{O} & \mathrm{N} & \#\mathrm{H} & \delta & q^* \end{array} \\ \begin{bmatrix} 1 & 0 & 0 & 3 & 1 & 0 \\ 1 & 0 & 0 & 1 & 3 & 0 \\ 1 & 0 & 0 & 3 & 1 & 0 \\ 1 & 0 & 0 & 2 & 2 & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ 1 & 0 & 0 & 0 & 3 & 0 \\ 0 & 1 & 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 1 & 1 & 0 \end{bmatrix} \end{array}$$

$$A = \begin{array}{c} \\ 1 \\ 2 \\ 3 \\ 4 \\ \vdots \\ 13 \\ 14 \\ 15 \end{array} \begin{array}{c} \begin{array}{cccccccc} 1 & 2 & 3 & 4 & & 13 & 14 & 15 \end{array} \\ \begin{bmatrix} 0 & 1 & 0 & 0 & \cdots & 0 & 0 & 0 \\ 1 & 0 & 1 & 1 & \cdots & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & \cdots & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & \cdots & 0 & 0 & 0 \\ \vdots & \vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \vdots \\ 0 & 0 & 0 & 0 & \cdots & 0 & 1 & 1 \\ 0 & 0 & 0 & 0 & \cdots & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & \cdots & 1 & 0 & 0 \end{bmatrix} \end{array}$$

Graph Neural Network

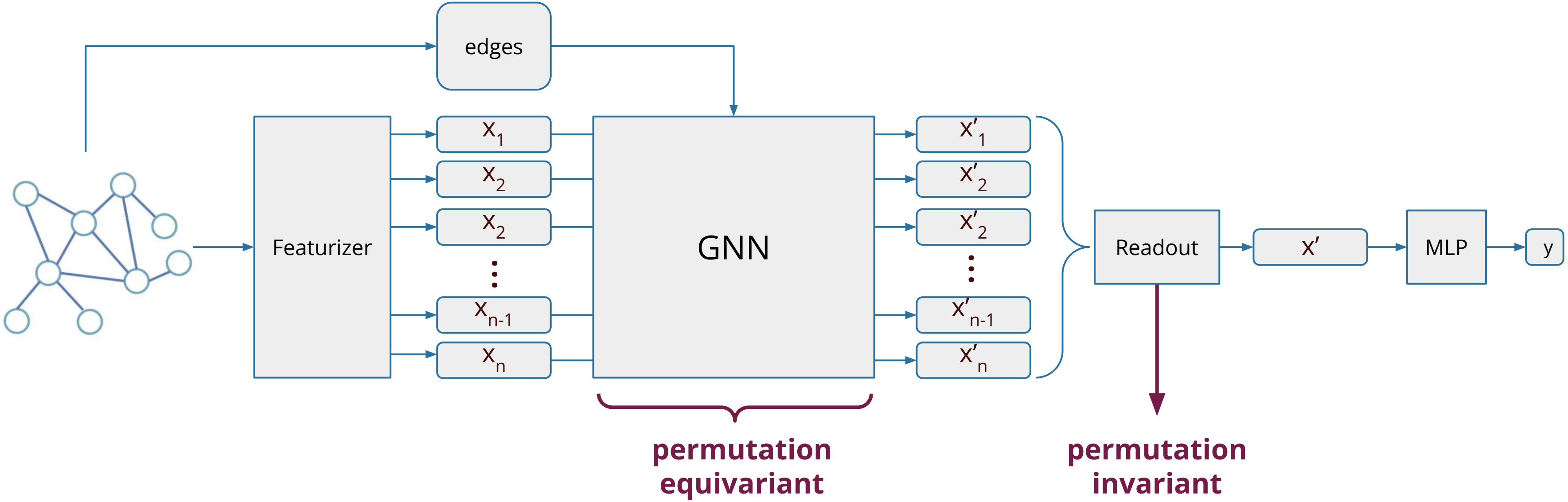


How shouldn't GNN look like?



How should GNN look like?

group of machine gmum learning research

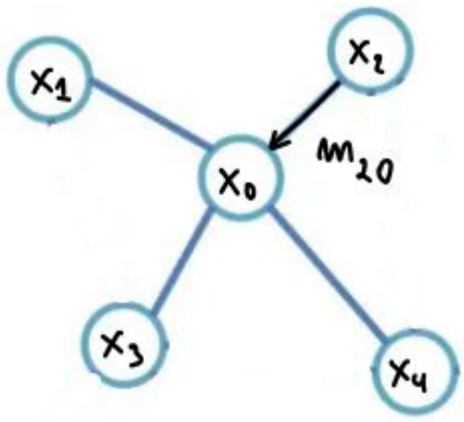


1. GNN should be permutation-equivariant.
2. GNN should deal with the graph structure given by edges.

Message Passing Neural Networks

Message Passing

Message Passing



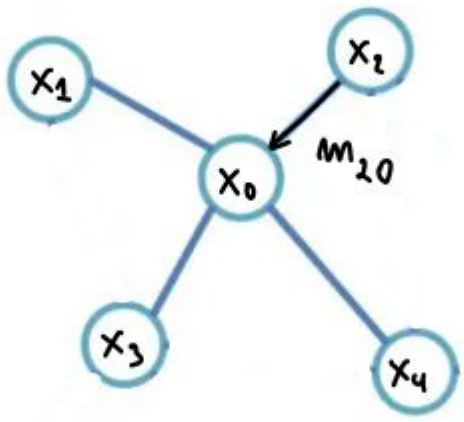
General:

$$m_{ji} = \psi(x_j, x_i)$$

Example:

$$m_{ji} = Wx_j$$

Message Passing



General:

$$m_{ji} = \psi(x_j, x_i)$$

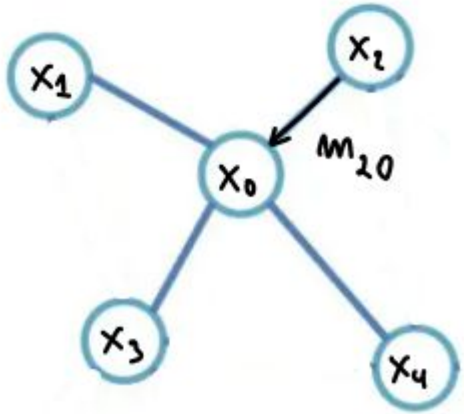
$$m_i = \square_{j \in N(i)} m_{ji}$$

Example:

$$m_{ji} = W x_j$$

$$m_i = \sum_{j \in N(i)} m_{ji}$$

Message Passing



General:

$$m_{ji} = \psi(x_j, x_i)$$

$$m_i = \square_{j \in N(i)} m_{ji}$$

$$x_i' = \rho(x_i, m_i)$$

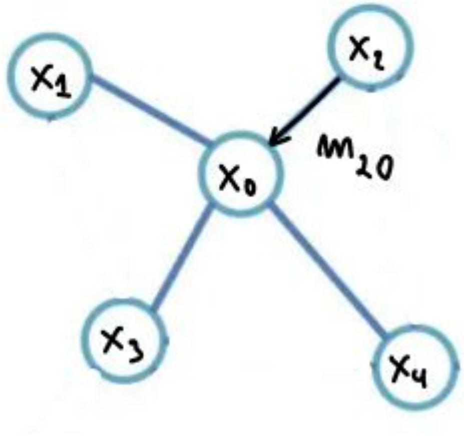
Example:

$$m_{ji} = W x_j$$

$$m_i = \sum_{j \in N(i)} m_{ji}$$

$$x_i' = W_1 x_i + W_2 m_i$$

Message Passing



General:

$$m_{ji} = \psi(x_j, x_i)$$

$$m_i = \square_{j \in N(i)} m_{ji}$$

$$x_i' = \rho(x_i, m_i)$$

$$x_i' = \rho(x_i, \square_{j \in N(i)} \psi(x_j, x_i))$$

Example:

$$m_{ji} = W x_j$$

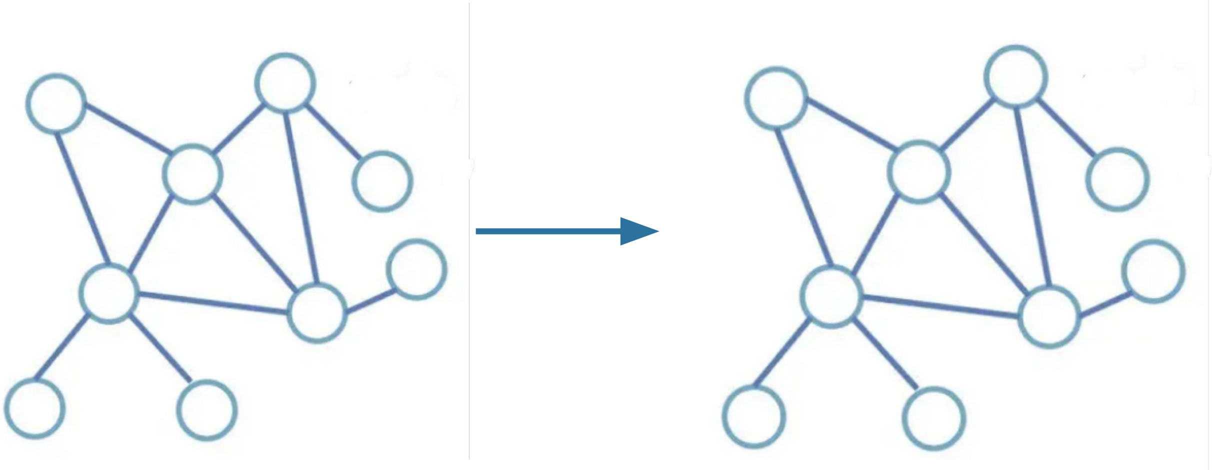
$$m_i = \sum_{j \in N(i)} m_{ji}$$

$$x_i' = W_1 x_i + W_2 m_i$$

$$x_i' = W_1 x_i + W_2 \sum_{j \in N(i)} W_3 x_j$$

Message Passing

Information Propagation

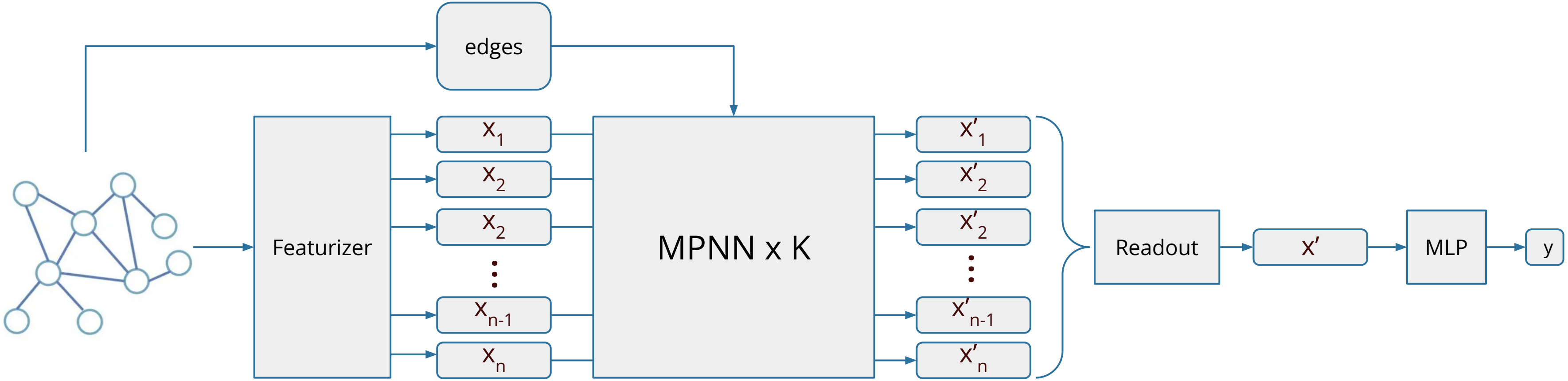


$(x_1^t, x_2^t, \ldots, x_n^t)$ $\quad\rightarrow\quad$ $(x_1^{t+1}, x_2^{t+1}, \ldots, x_n^{t+1})$

$$x_i^{t+1} = \rho(x_i^t, \square_{j \in N(i)} \psi(x_j^t, x_i^t))$$

Message Passing

Is it a proper GNN?



1. MPNN is permutation-equivariant.
2. MPNN deals with the graph structure given by edges.

Message Passing

Examples

group of machine gmum learning research

GIN:

$$\mathbf{x}'_i = h_{\mathbf{\Theta}} \left((1 + \epsilon) \cdot \mathbf{x}_i + \sum_{j \in \mathcal{N}(i)} \mathbf{x}_j \right)$$

GCN:

$$\mathbf{x}'_i = \mathbf{\Theta}^{\top} \sum_{j \in \mathcal{N}(i) \cup \{ i \}} \frac{e_{j,i}}{\sqrt{\hat{d}_j \hat{d}_i}} \mathbf{x}_j$$

GraphSAGE:

$$\mathbf{x}'_i = \mathbf{W}_1 \mathbf{x}_i + \mathbf{W}_2 \cdot \mathrm{mean}_{j \in \mathcal{N}(i)} \mathbf{x}_j$$

GAT:

$$\mathbf{x}'_i = \alpha_{i,i}\mathbf{\Theta}_{s}\mathbf{x}_{i} + \sum_{j \in \mathcal{N}(i)} \alpha_{i,j}\mathbf{\Theta}_{t}\mathbf{x}_{j},$$

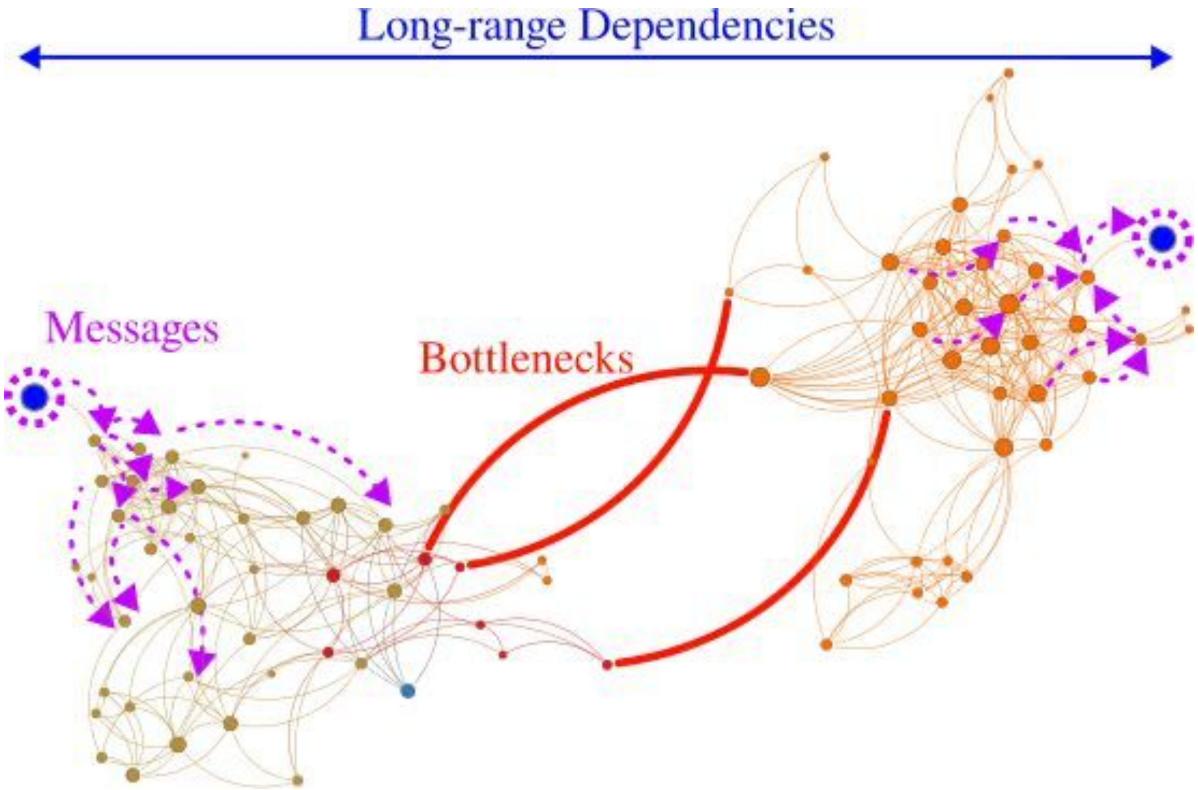
Message Passing

Issues

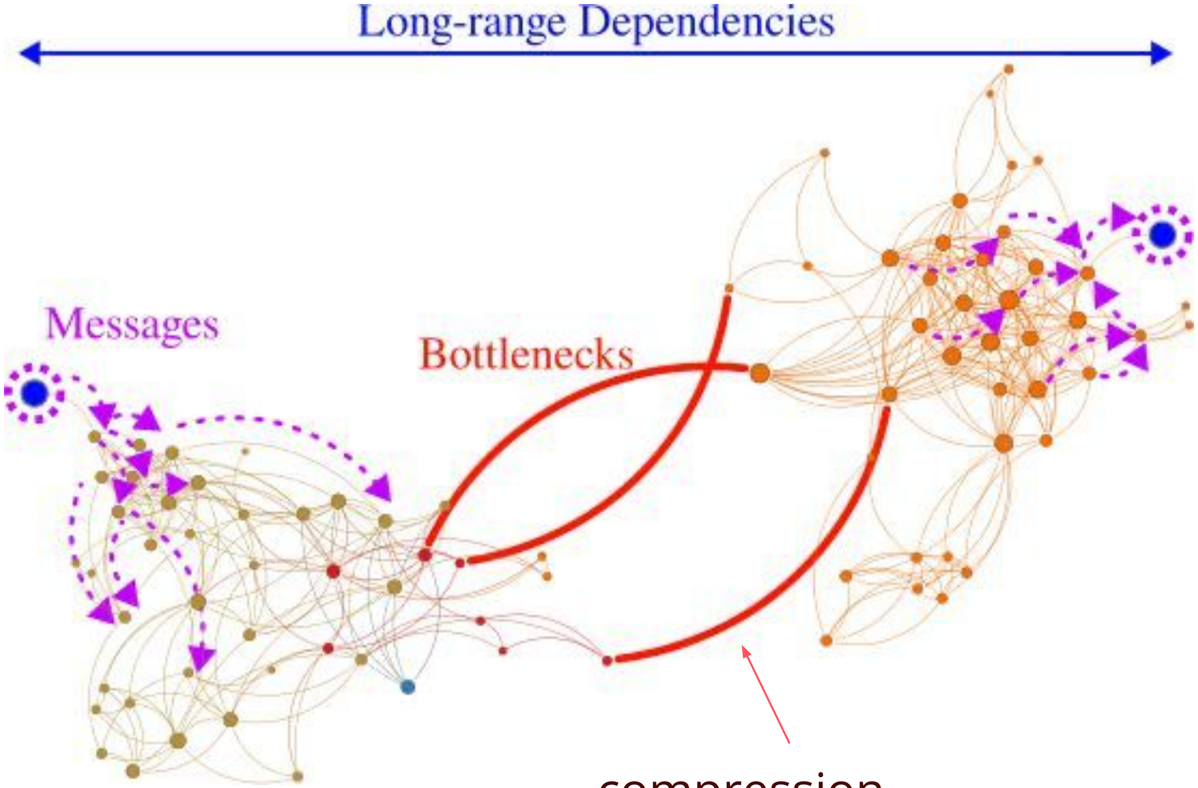
What are the issues?

Long-range dependencies

group of machine gmum learning research

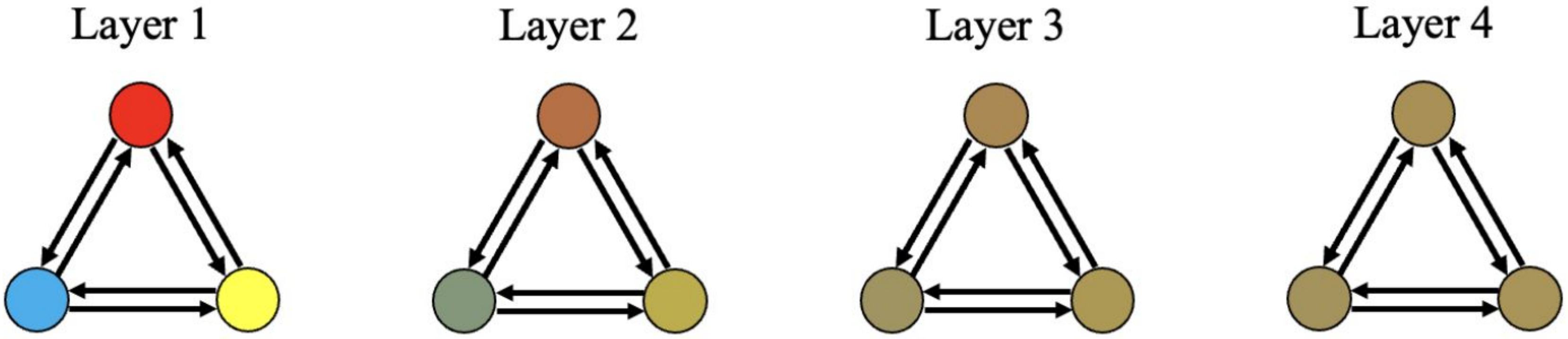


Oversquashing

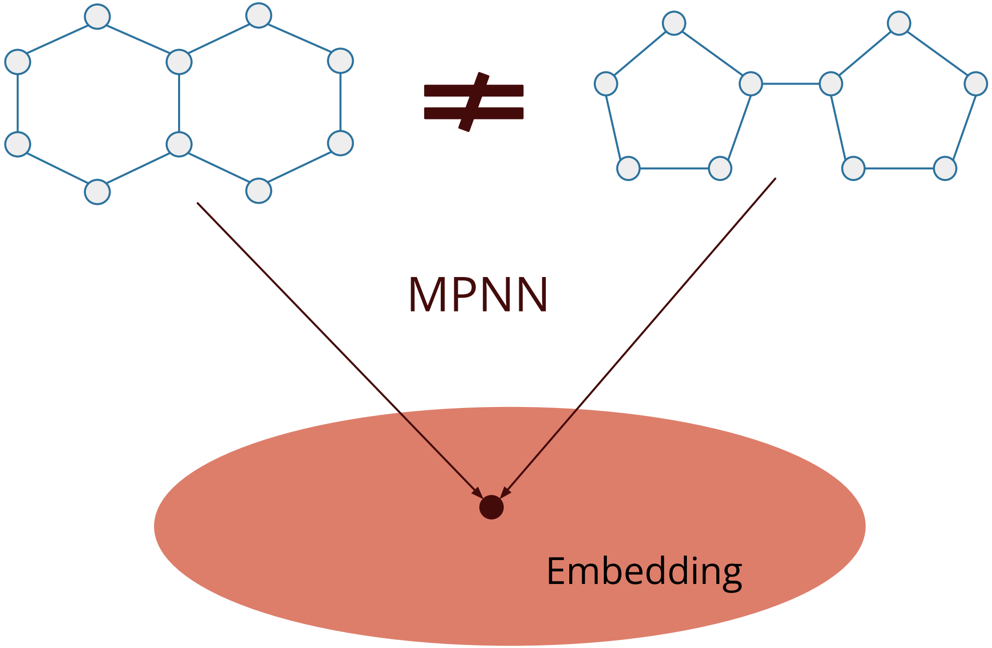


compression

Oversmoothing



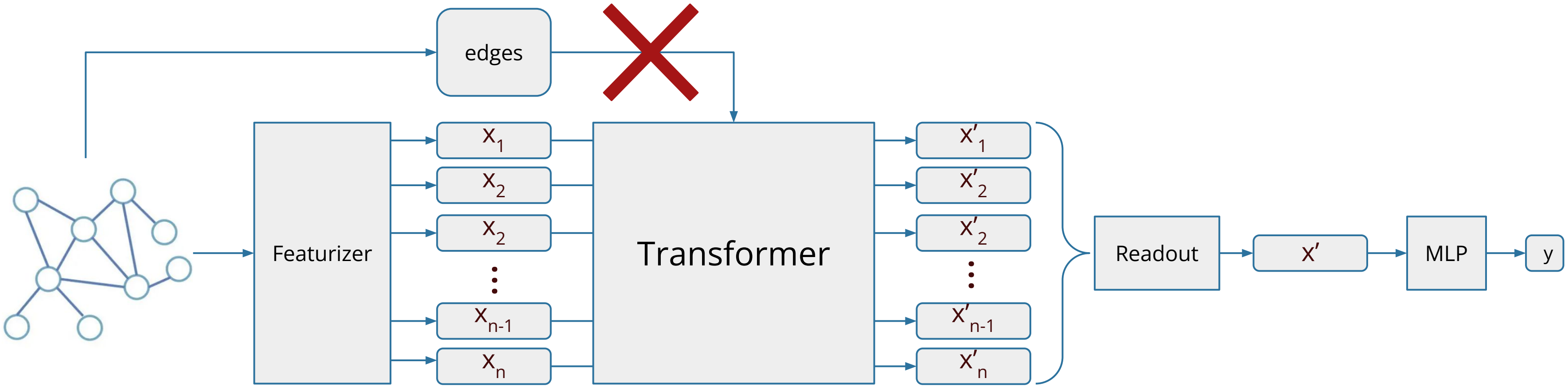
Expressivity



Transformers as GNNs

Transformers

Is it a proper GNN?

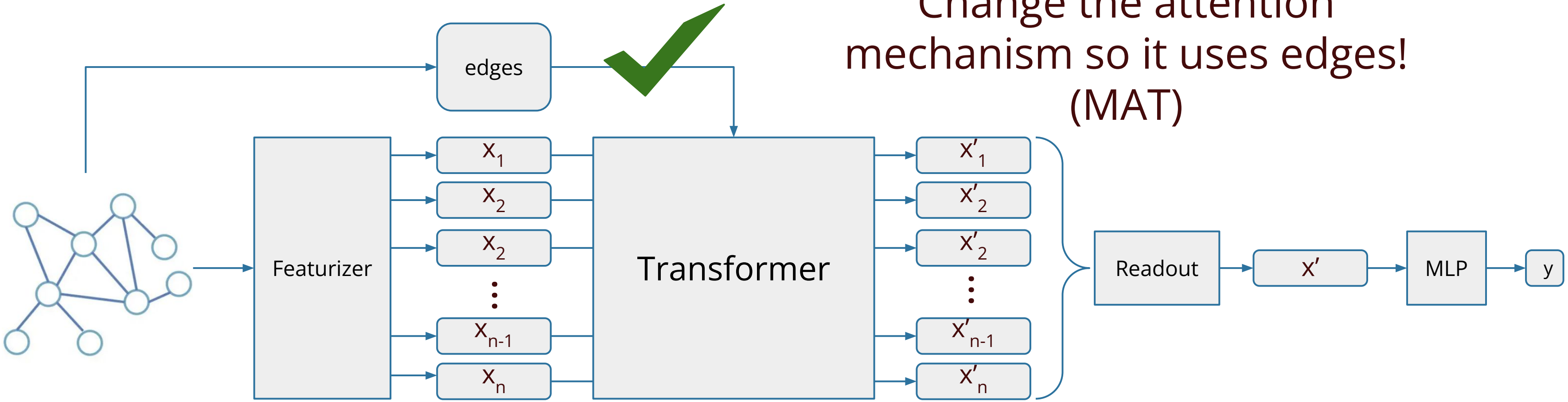


1. Transformer is permutation-equivariant.
2. Transformer cannot deal with the graph structure given by edges by default :<

Transformers

Is it a proper GNN?

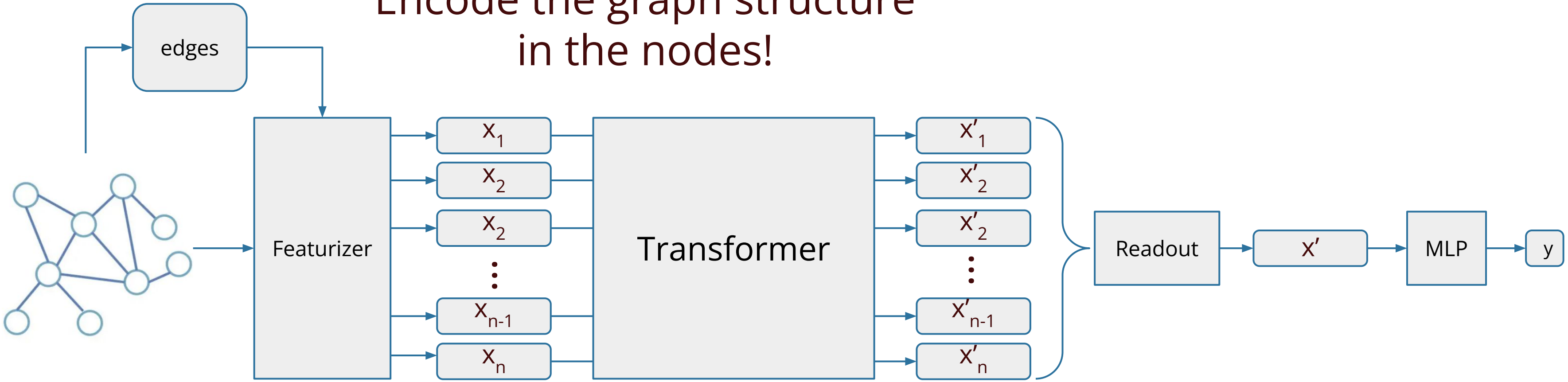
Change the attention mechanism so it uses edges! (MAT)



Transformers

Is it a proper GNN?

Encode the graph structure
in the nodes!



1. It can be done with structural/positional encodings (e.g. Random Walk).
2. Or it can be than with MPNN! (GraphGPS)

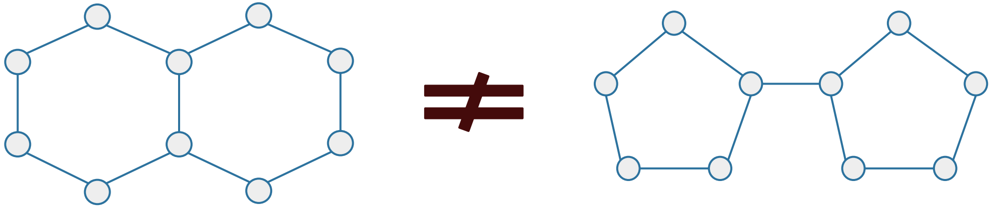
Transformers

Issues

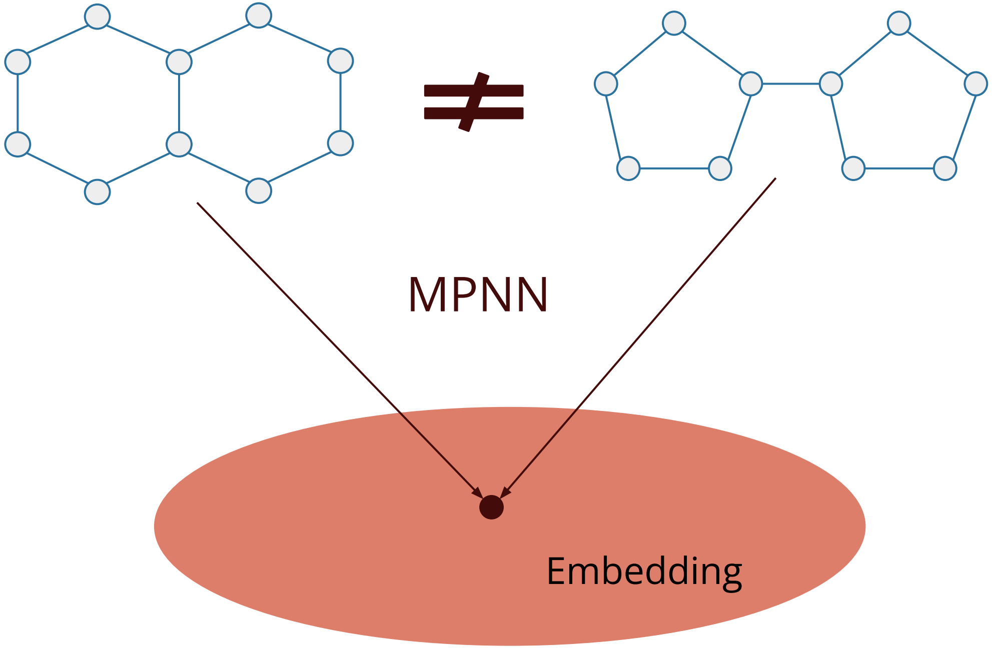
Transformers deal with long-range dependencies and oversquashing.

Expressivity of Message Passing

Can MPNN distinguish all graphs?



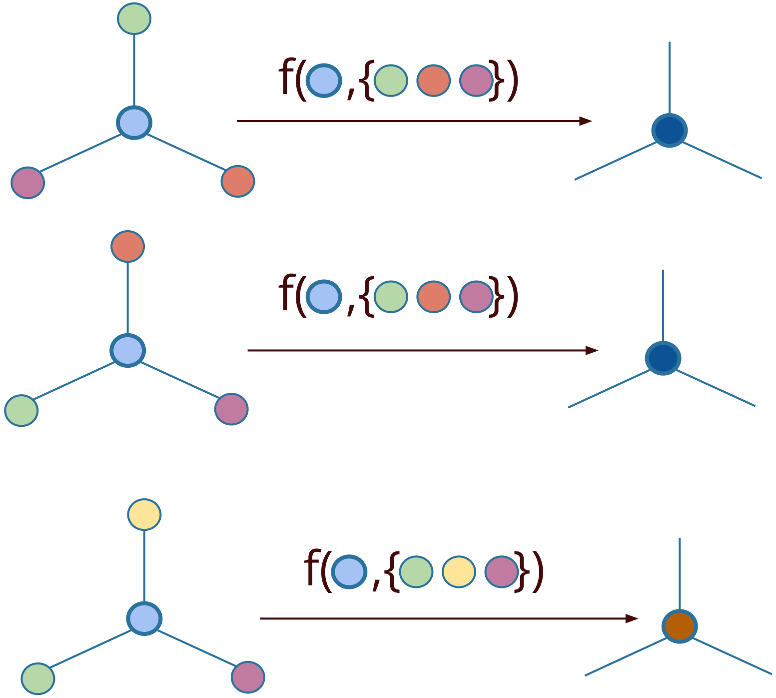
MPNN cannot distinguish all graphs!



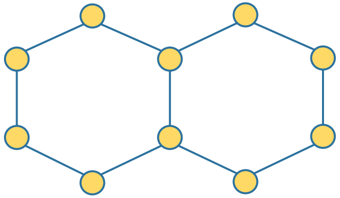
The most powerful MPNN

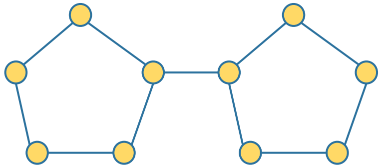
group of machine learning research

- Let us assume that all nodes in a graph have the same initial encodings.
- Let us denote node embeddings as colors. Different color -> different embedding.
- MPNN can only return different colors for nodes with different neighborhood.
- Our coloring MPNN always returns different color for nodes with different neighbors.



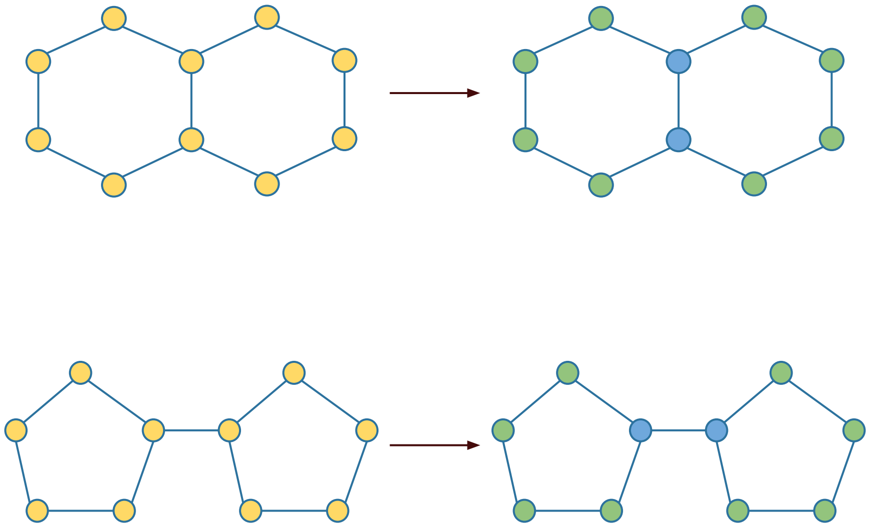
Coloring MPNN



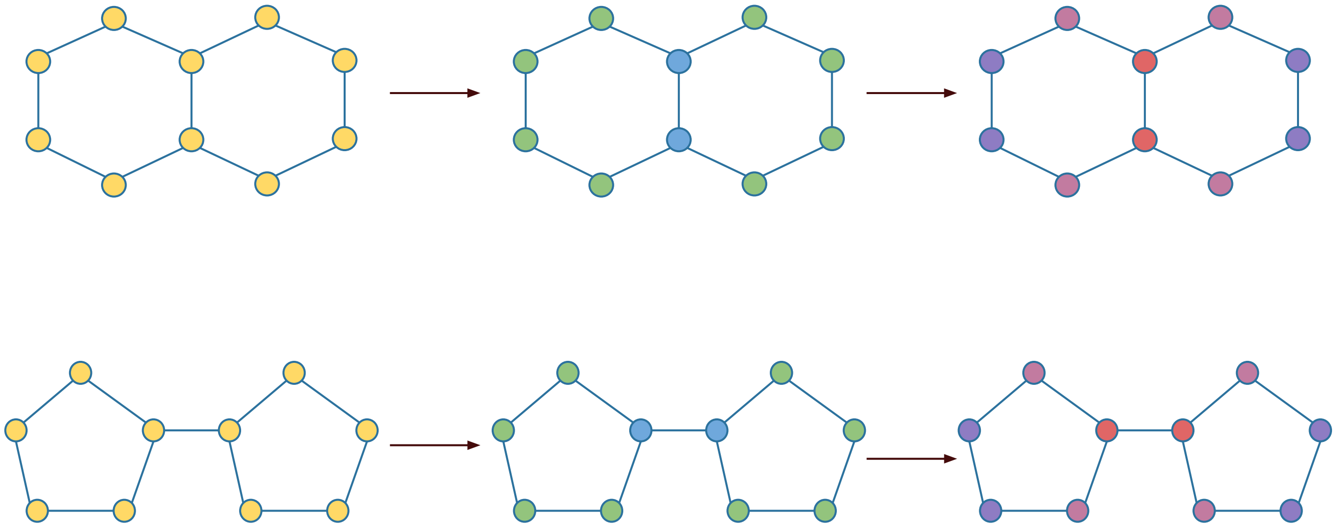


Coloring MPNN

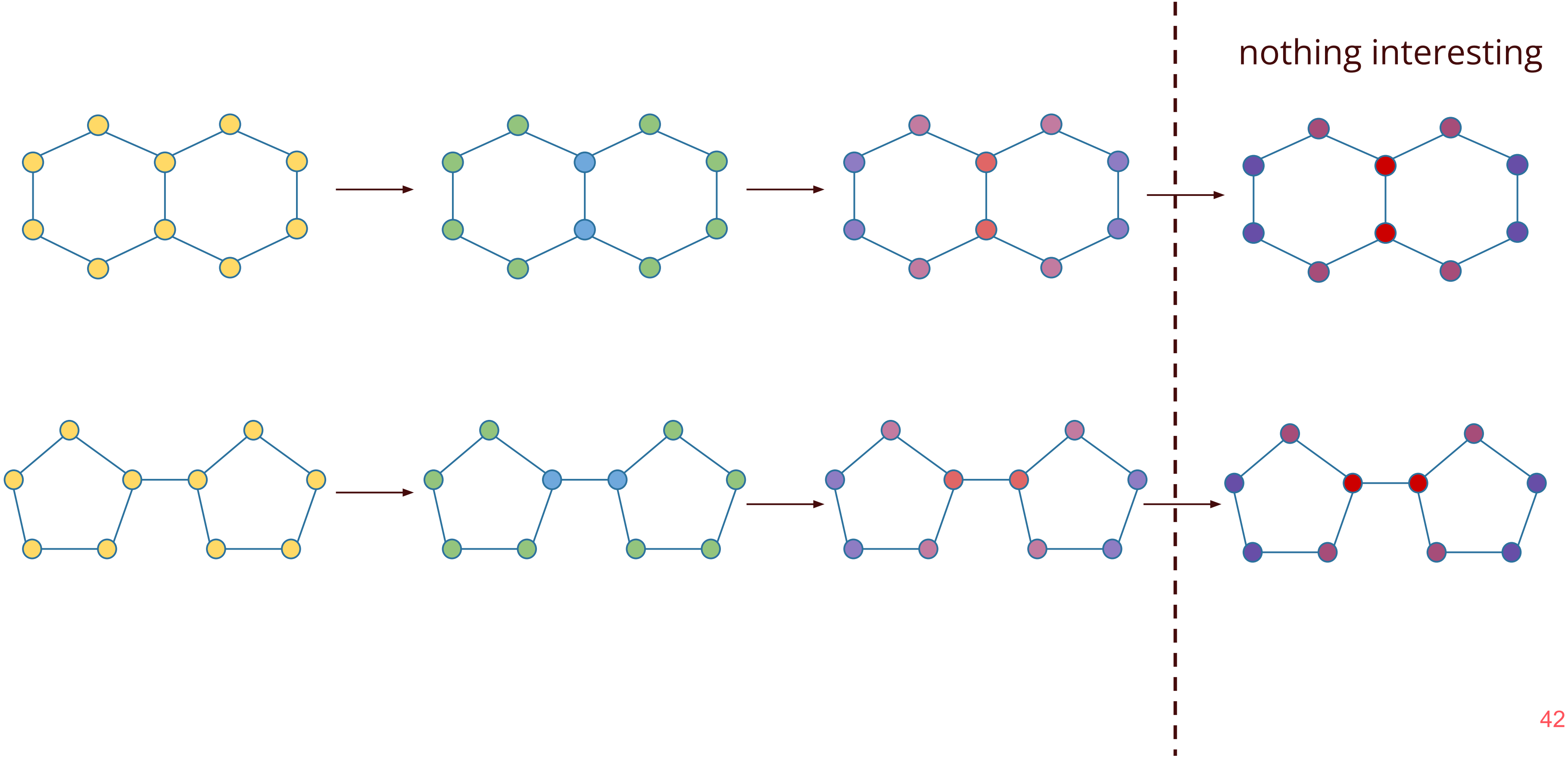
group of machine
gmum
learning research



Coloring MPNN

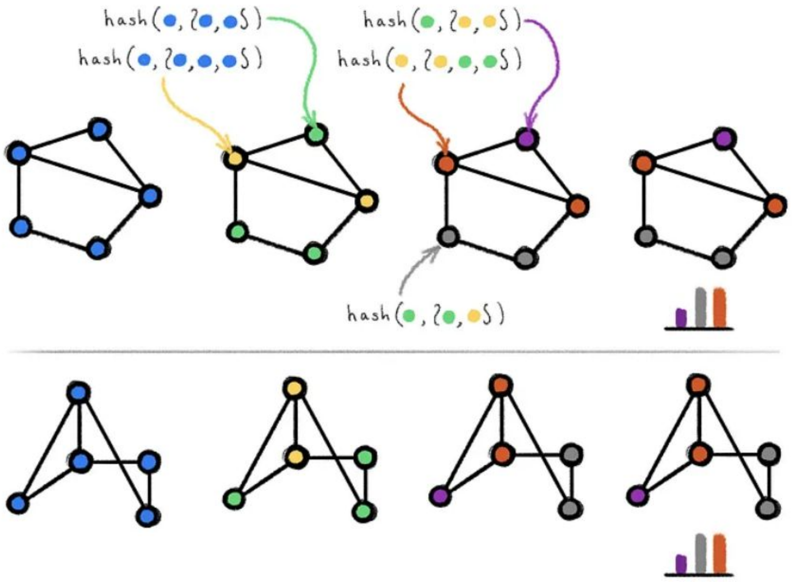


Coloring MPNN



WL test

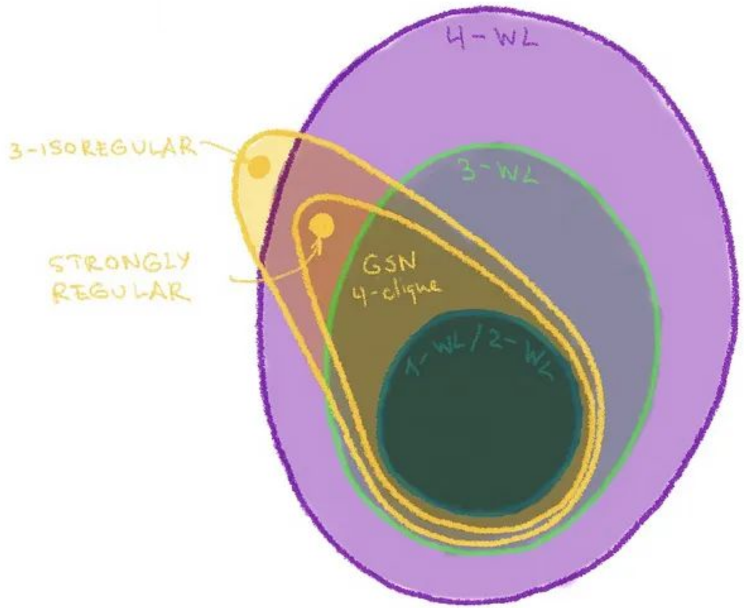
- Our coloring MPNN works very similar to Weisfeiler-Lehman graph isomorphism test.
- No MPNN is more powerful than WL test.



k-WL test

group of machine gmum learning research

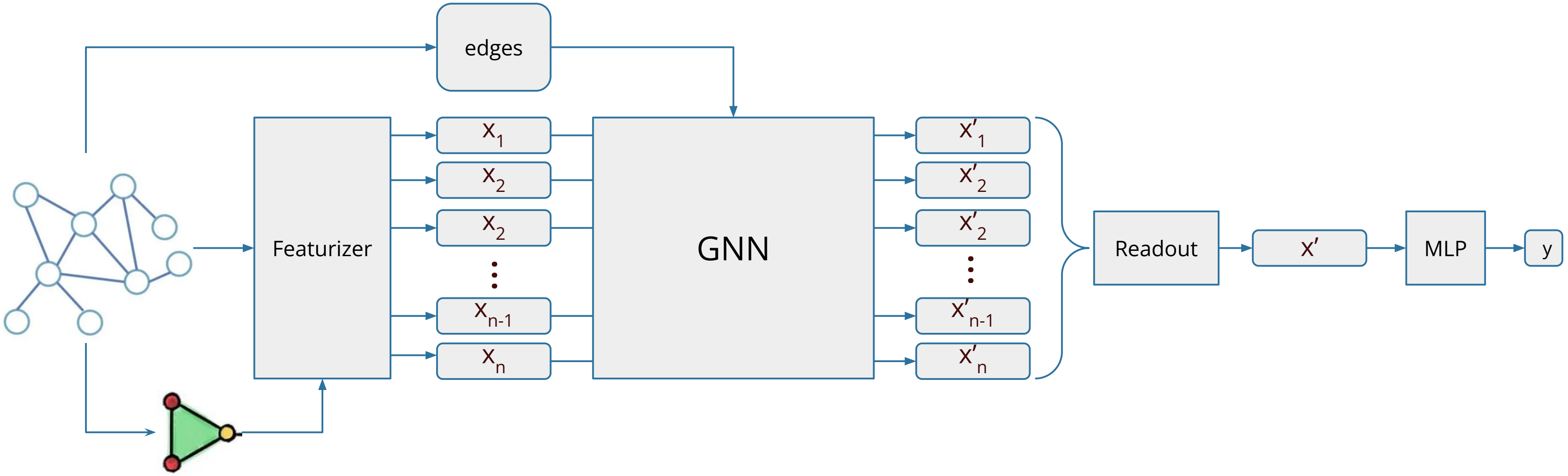
- We can easily generalize the WL test and obtain k-WL test. (k+1)-WL test is strictly more powerful than k-WL test (for k>1).
- There is k-GNN model which mimics the k-WL test and is O(n^k).
- There is an IGN model as powerful as 3-WL and O(n^2).
- But we can escape the k-WL classification...



Beyond k-WL classifiation

Examples of non-isomorphic graphs that cannot be distinguished by 1-WL but can be distinguished by 3-WL due to its capability of counting triangles.

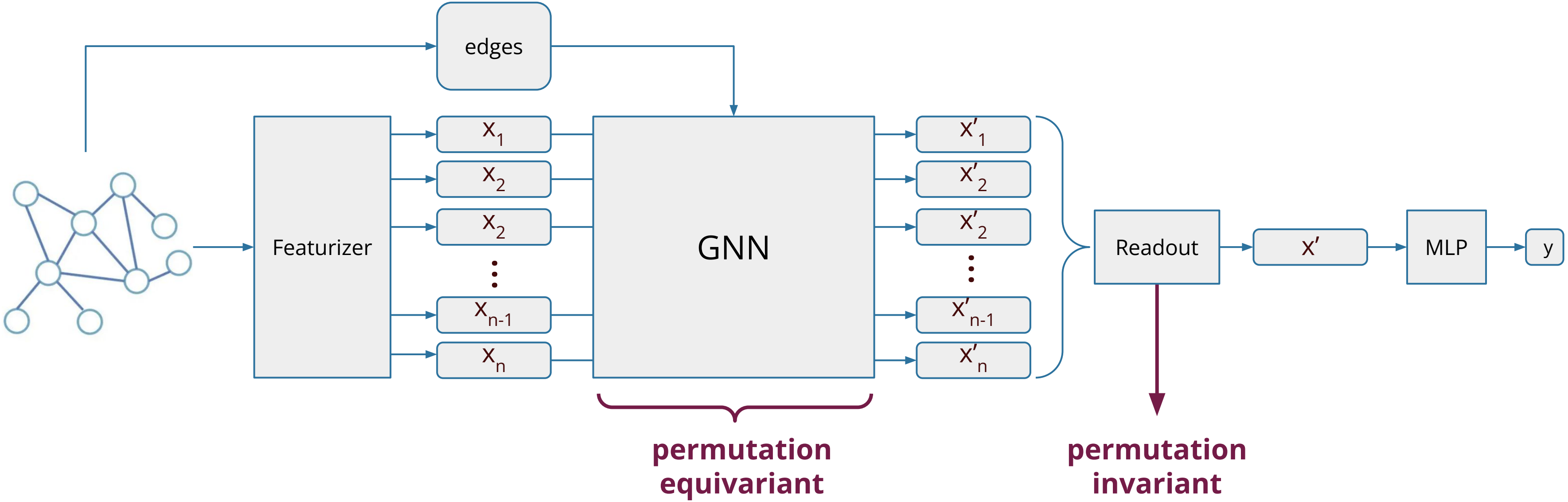
Beyond k-WL classifiation



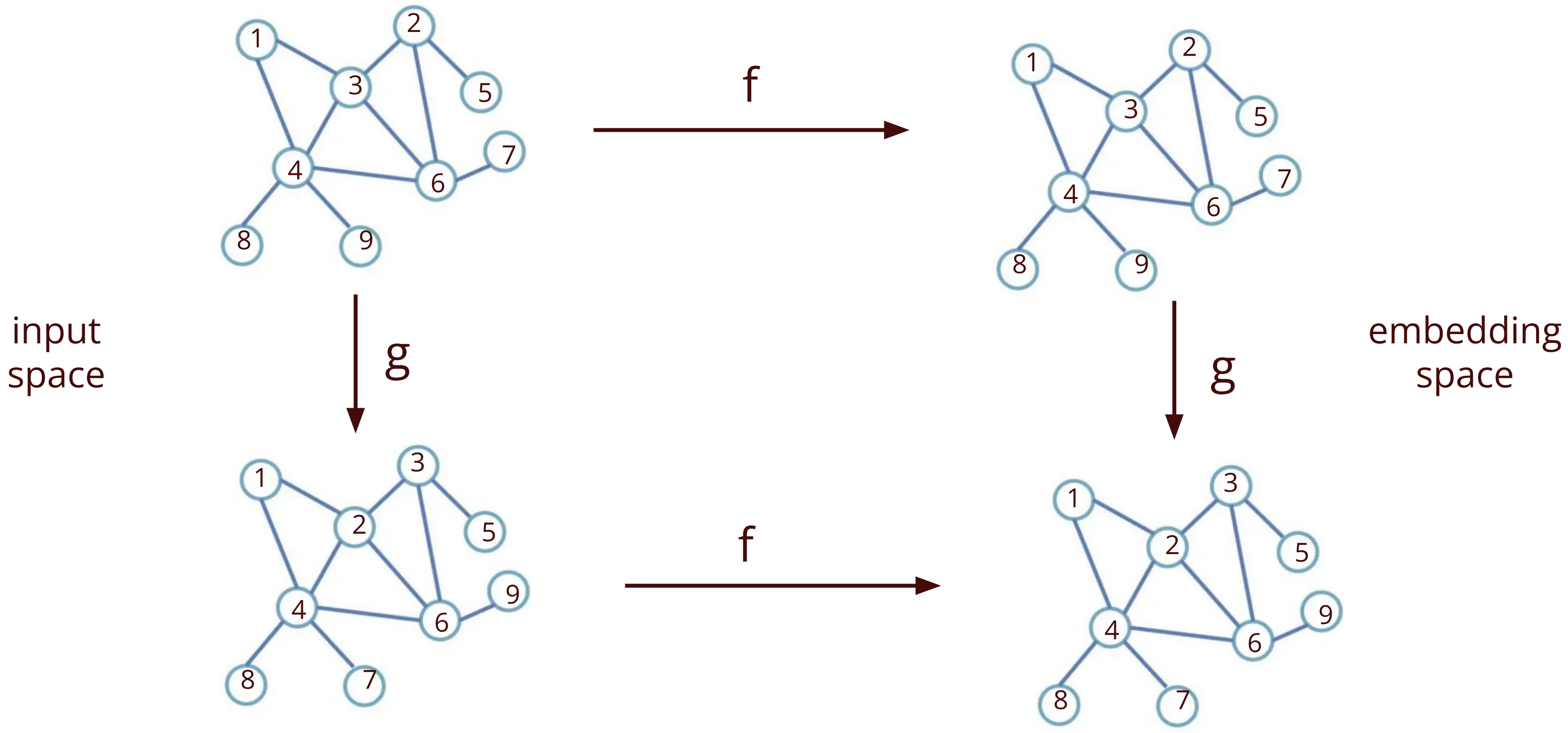
We can simply count triangles and
enrich the node encodings!

Symmetries: Equivariant Deep Learning

Permutation Equivariance

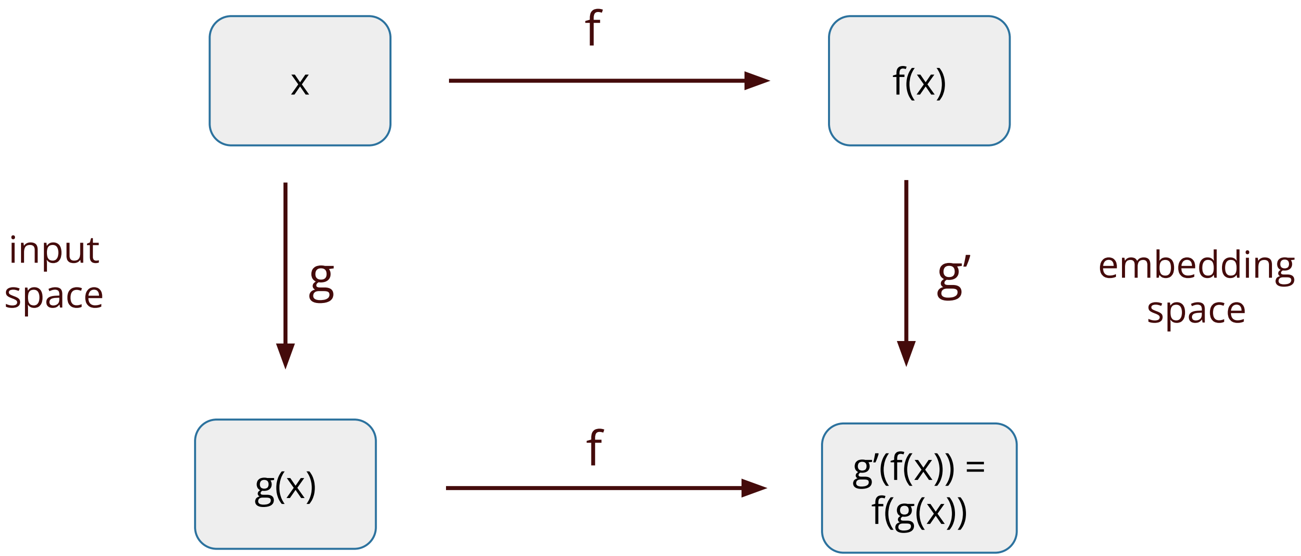


Permutation Equivariance



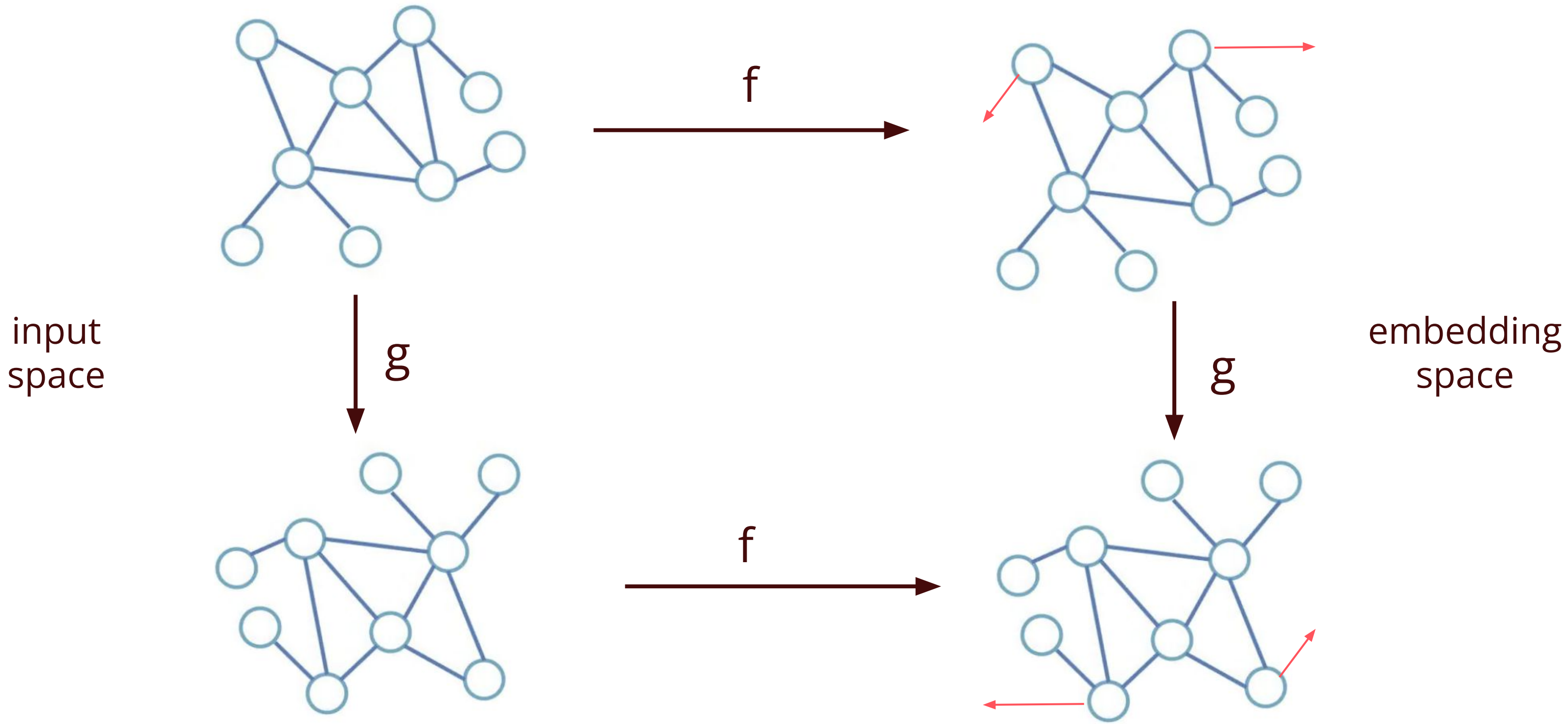
Equivariance

f - G-equivariant function
g - operation from group G defined in input space
g’ - operation from group G’ defined in embedding space corresponding to G



Rotation Equivariance

group of machine learning research



Deep Equivariant Learning

- Some node-level prediction tasks requires equivariance with respect to e.g. rotations.
- Equivariance can be used to incorporate invariance.
- Deep Equivariant Learning is a fast growing field. It definitely requires its own course.

GNNs are great!

Recommended course

group of machine learning research

Machine Learning in Drug Discovery

by Sabina Smusz and Tomasz Danel

Inspirations

group of machine
gmum
learning research

LoGG reading group: yt channel

Geometric Deep Learning: The Erlangen Programme of ML: video

Group Equivariant Deep Learning: yt series

Machine Learning in Drug Discovery: UJ lectures and labs

Machine Learning with Graphs course from Stanford: webpage

Worth reading

- MPNNs:
  - GAT: Graph Attention Networks
  - GraphSAGE: Inductive Representation Learning on Large Graphs
  - GIN: How Powerful are Graph Neural Networks?
- Transformers:
  - MAT: Molecule Attention Transformer (GMUM)
  - GraphGPS: Recipe for a General, Powerful, Scalable Graph Transformer
  - SAN: Rethinking Graph Transformers with Spectral Attention
- k-WL GNNs:
  - k-GNN: Weisfeiler and Leman go neural: Higher-order graph neural networks

  - IGN: Convergence of Invariant Graph Networks

Worth reading

- Geometric GNNs:
  - SGCN: Spatial Graph Convolutional Networks (GMUM)
  - Geometric Transformer: Geometric Transformer for End-to-End Molecule Properties Prediction
- Equivariant GNNs:
  - EGNN: E(n) Equivariant Graph Neural Networks
  - SE(3)-Transformers: 3D Roto-Translation Equivariant Attention Networks
- Symmetry-breaking GNNs:
  - ChIRo: Learning 3D Representations of Molecular Chirality with Invariance to Bond Rotations
  - ChiENN: Embracing Molecular Chirality with Graph Neural Networks (GMUM)
- Random:
  - Understanding convolution on graphs via energies

Thanks for your attention!