How powerful are Graph Neural Networks?

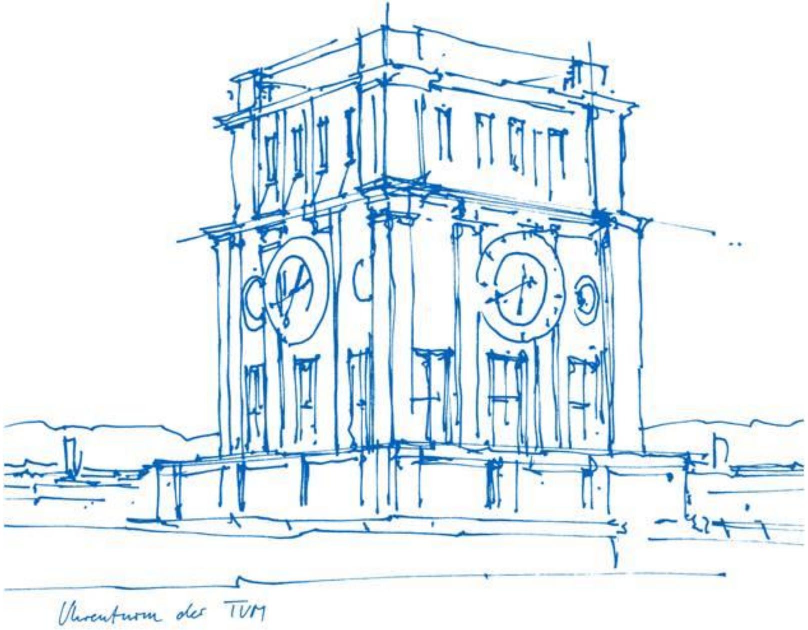
Vincent Limbach

Technical University of Munich

TUM School of Computation, Information and Technology

Computer Vision Group

Garching, 09. October 2023

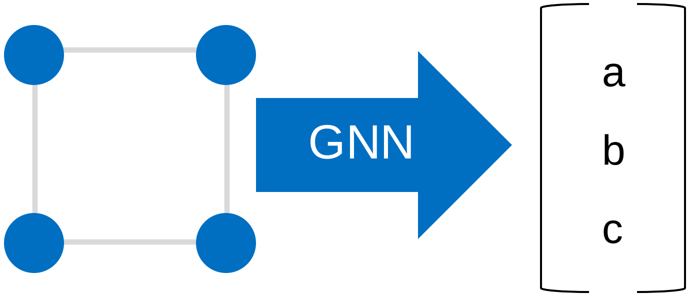


Recap of Graph Neural Networks (GNNs)

State-of-the-art performance in various fields

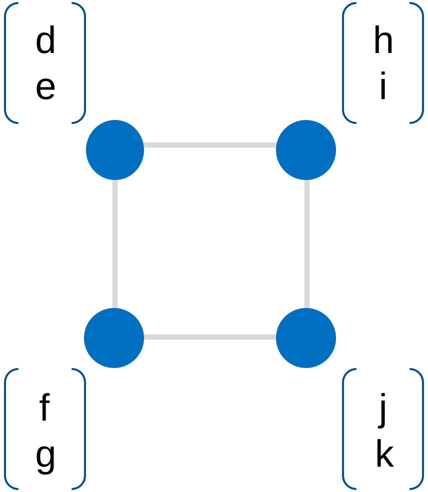
Recap of Graph Neural Networks (GNNs)

State-of-the-art performance in various fields



$$\mathcal{G} \to \mathbb{R}^d$$

Recap of Graph Neural Networks (GNNs)

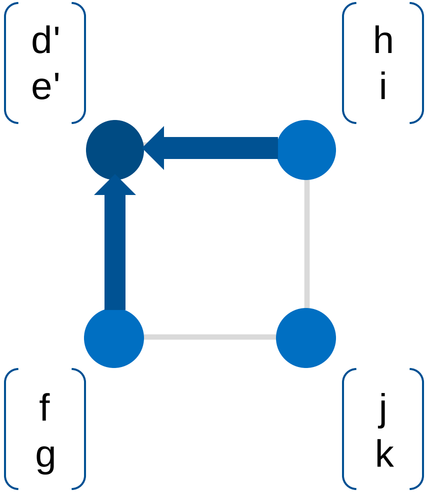


State-of-the-art performance in various fields

A GNN maps a graph to a vector:

- Embed information in numerical vectors (labels)

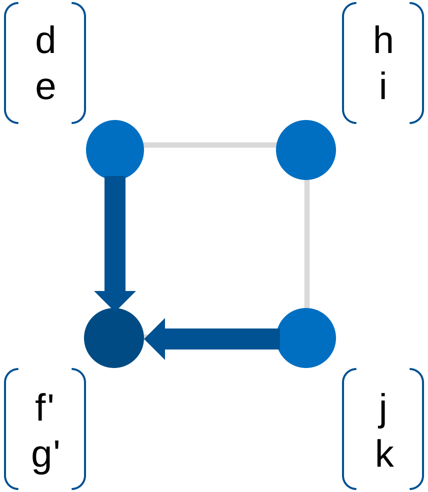
Recap of Graph Neural Networks (GNNs)



State-of-the-art performance in various fields

A GNN maps a graph to a vector:
- Embed information in numerical vectors (labels)
- Aggregate neighbouring labels for new label

Recap of Graph Neural Networks (GNNs)



State-of-the-art performance in various fields

A GNN maps a graph to a vector:

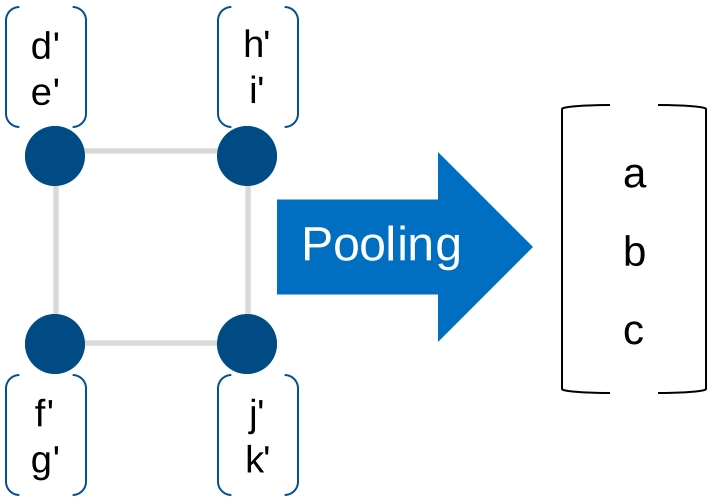
- Embed information in numerical vectors (labels)
- Aggregate neighbouring labels for new label

Recap of Graph Neural Networks (GNNs)

State-of-the-art performance in various fields

A GNN maps a graph to a vector:

- Embed information in numerical vectors (labels)
- Aggregate neighbouring labels for new label
- Aggregate all node labels to a vector in final layer



How powerful are GNNs?

How powerful can GNNs theoretically be?

What properties determine the expressiveness of a GNN?

How powerful can GNNs theoretically be?

How powerful can GNNs theoretically be?

Can GNNs solve the graph isomorphism problem?

Graph Isomorphism Problem

No algorithm with polynomial runtime known

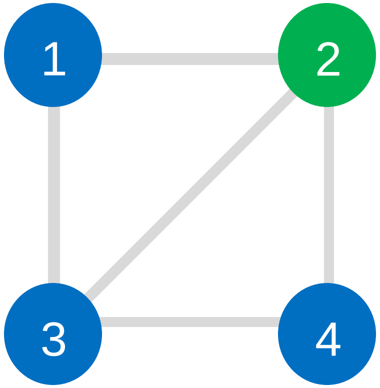
If GNNs could distinguish all graphs, then we would have a polynomial solution!

Weisfeiler-Lehman (WL) Test

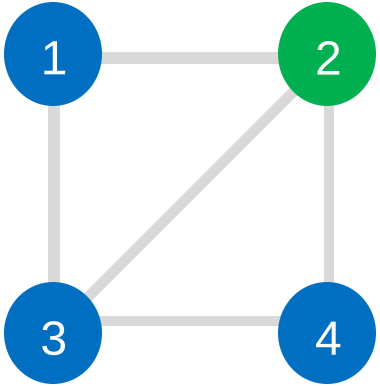
Classical isomorphism test based on labels

Powerful but not capable to distinguish all graphs

WL Test



WL Test



T = 0

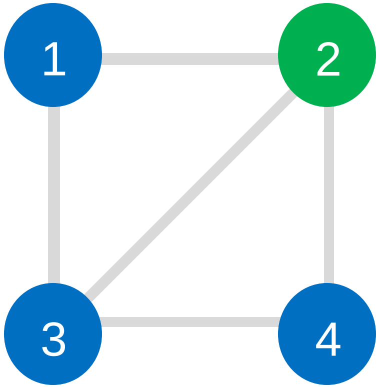
1:{B}

2:{G}

3:{B}

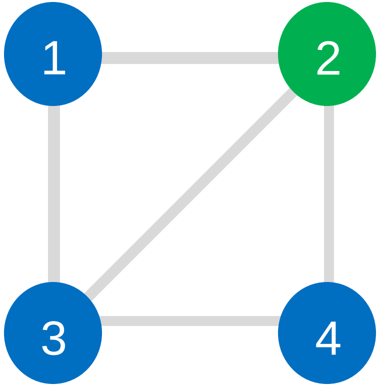
4:{B}

WL Test



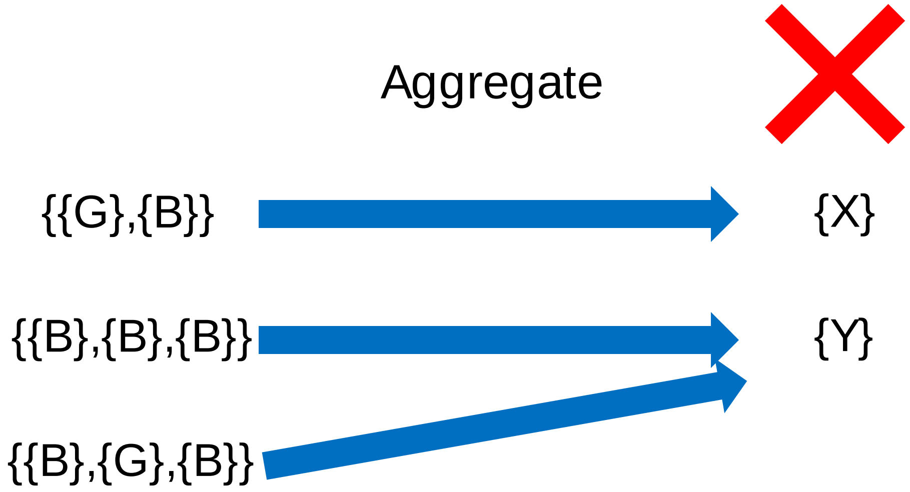
T = 0	T = 1
1:{B}	1:{{G},{B}}
2:{G}	2:{{B},{B},{B}}
3:{B}	3:{{B},{G},{B}}
4:{B}	4:{{G},{B}}

WL Test

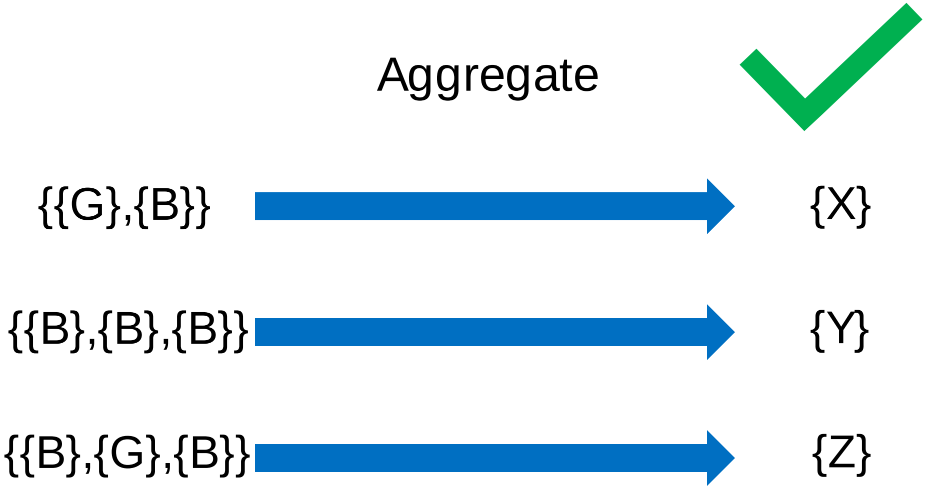


T = 0	T = 1	T = 2
1:{B}	1:{{G},{B}}	1:{{{B},{B},{B}},
2:{G}	2:{{B},{B},{B}}	{{B},{G},{B}}}
3:{B}	3:{{B},{G},{B}}	2: ...
4:{B}	4:{{G},{B}}	...

Can GNNs be as powerful as WL?



Can GNNs be as powerful as WL?



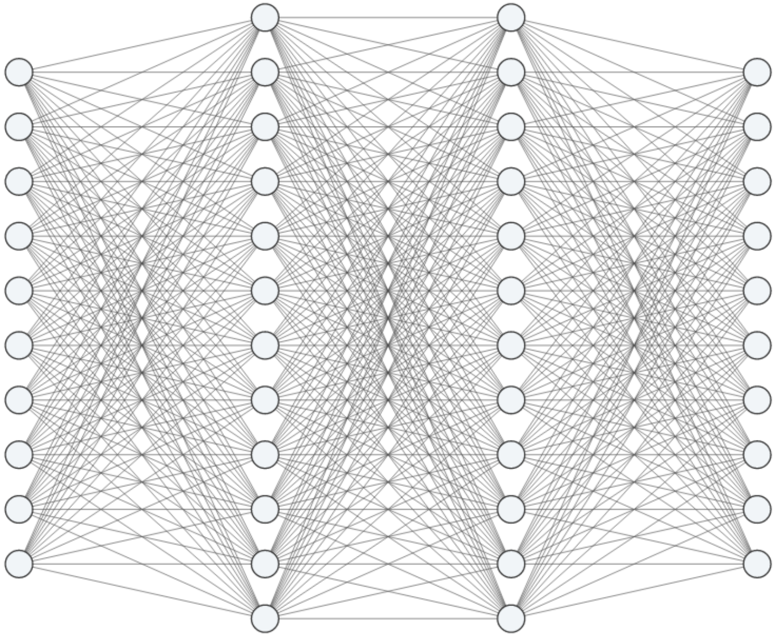
Universal Approximation Theory

Universal approximation theorem
An MLP with a linear output layer and one hidden layer can approximate any continuous function defined over a closed and bounded subset of $\mathbb{R}^D$, under mild assumptions on the activation function ('squashing' activation functions; e.g. sigmoid) and given the number of hidden units is large enough.

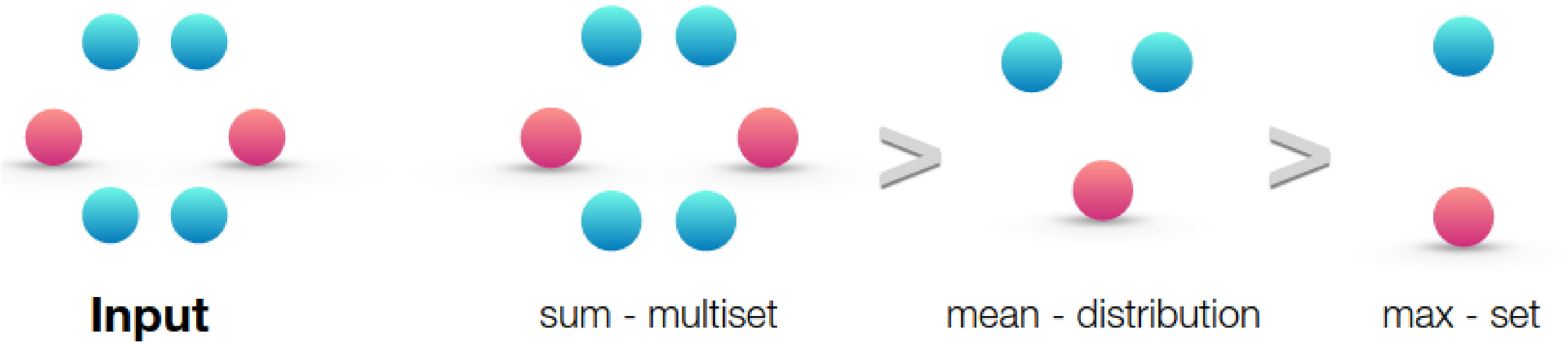
[Cybenko 1989; Funahashi 1989; Hornik et al 1989, 1991; Hartman et al 1990].

Universal Approximation Theory

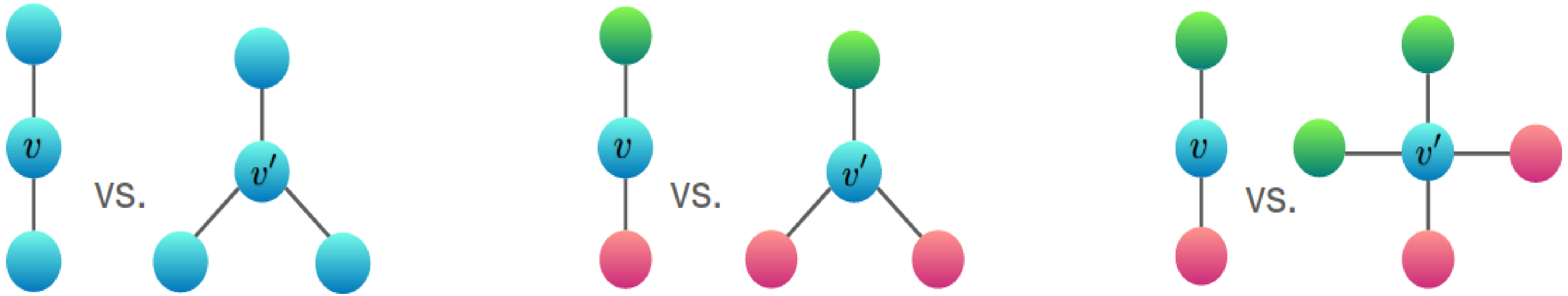
$$f(x) \equiv$$



What properties determine the expressiveness of a GNN?



What properties determine the expressiveness of a GNN?



(a) Mean and Max both fail

$$\begin{pmatrix}1\\0\\0\end{pmatrix}\begin{pmatrix}1\\0\\0\end{pmatrix} \quad \begin{pmatrix}1\\0\\0\end{pmatrix}\begin{pmatrix}1\\0\\0\end{pmatrix}$$

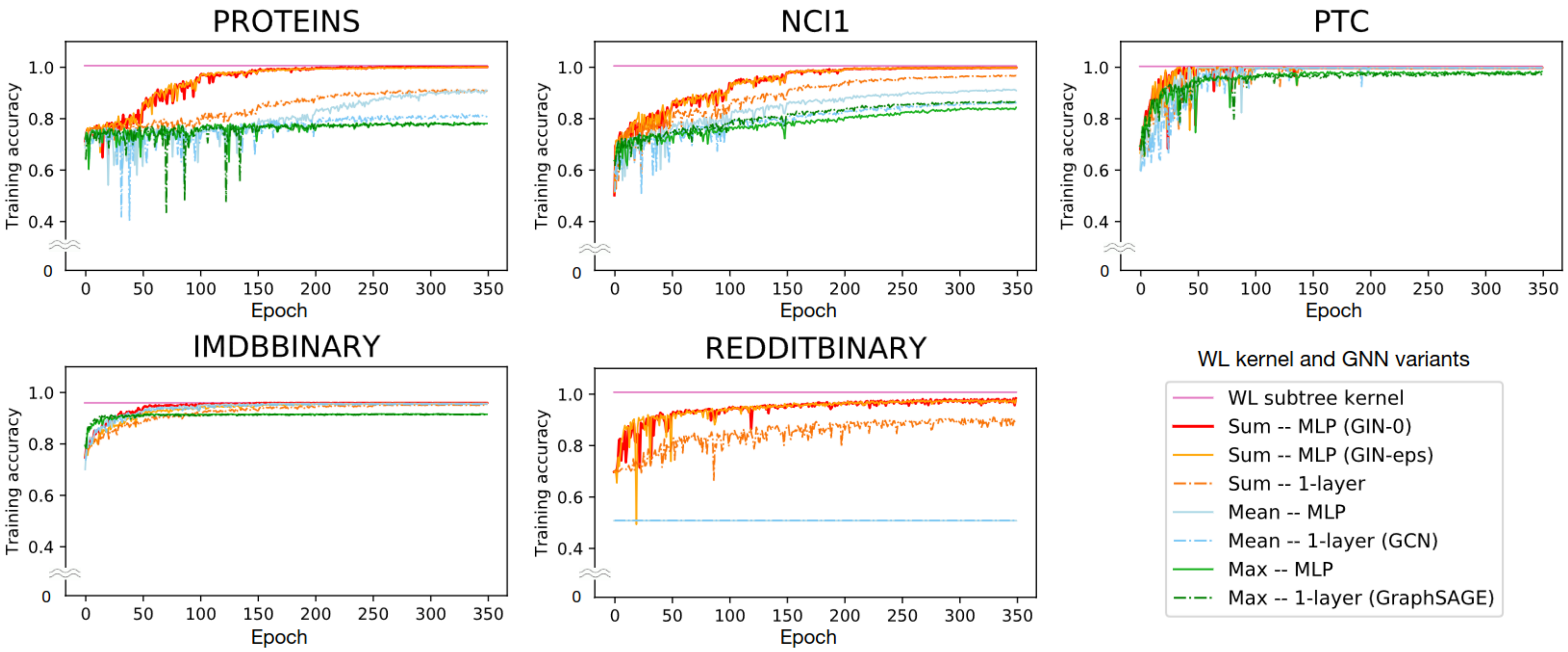
(b) Max fails

$$\begin{pmatrix}0\\1/2\\1/2\end{pmatrix}\begin{pmatrix}0\\1\\1\end{pmatrix} \quad \begin{pmatrix}0\\2/3\\1/3\end{pmatrix}\begin{pmatrix}0\\1\\1\end{pmatrix}$$

(c) Mean and Max both fail

$$\begin{pmatrix}0\\1/2\\1/2\end{pmatrix}\begin{pmatrix}0\\1\\1\end{pmatrix} \quad \begin{pmatrix}0\\2/4\\2/4\end{pmatrix}\begin{pmatrix}0\\1\\1\end{pmatrix}$$

Emperical Verification



How should we address these limitations?

1. Expand and modify GNN architecture
2. Design and test different architectures
3. Increase label information

Summary

1 GNNs can be as powerful as WL Test

2 The information lost by transformations and aggregations determines the capabilities of a GNN

3 Non-injective transformations and aggregations lead to decreased performance

Do you have any questions?

Thank you for your attention