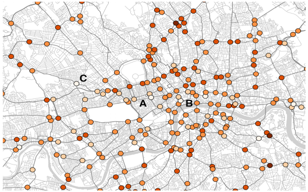
Graph Attention Networks

Justin Pauckert

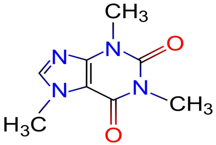
Discrete Optimization and Machine Learning
TU Berlin

July 3, 2021

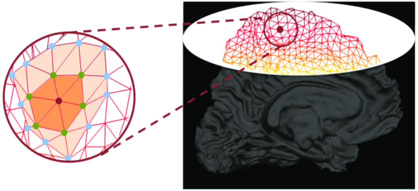
Graph Data



Traffic



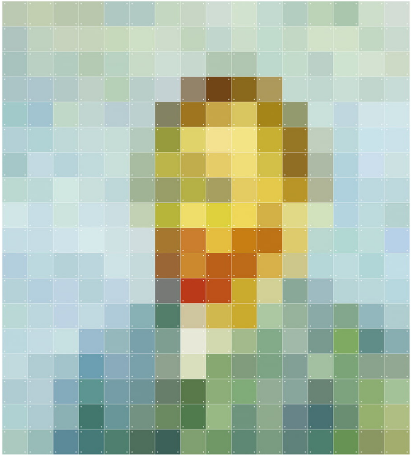
Chemistry



Neuroscience

Problem Setup

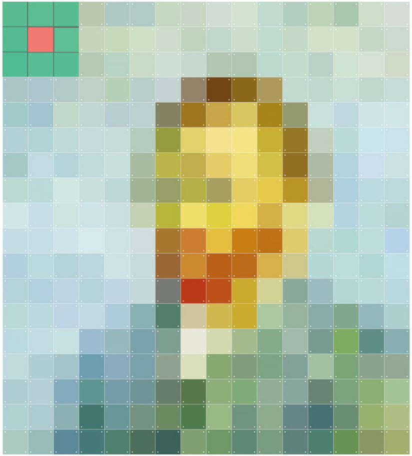
Grid Data



Graph Data

Problem Setup

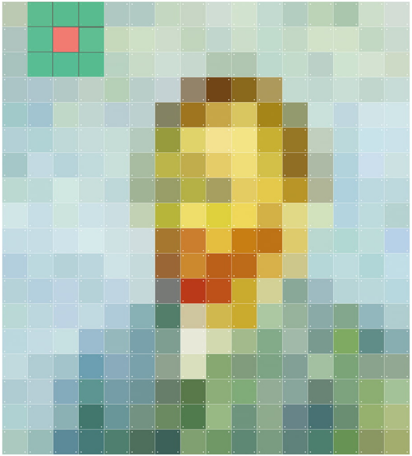
Grid Data



Graph Data

Problem Setup

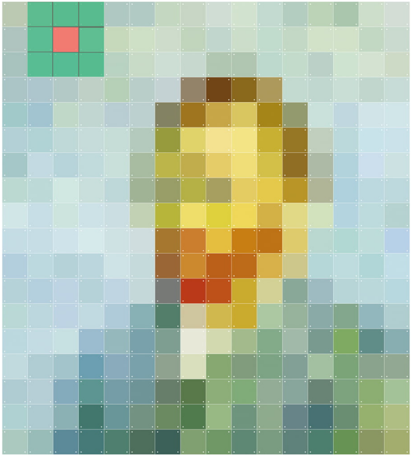
Grid Data



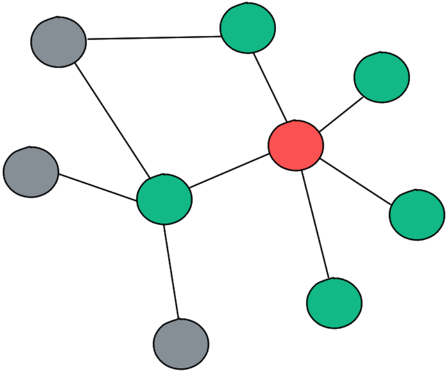
Graph Data

Problem Setup

Grid Data

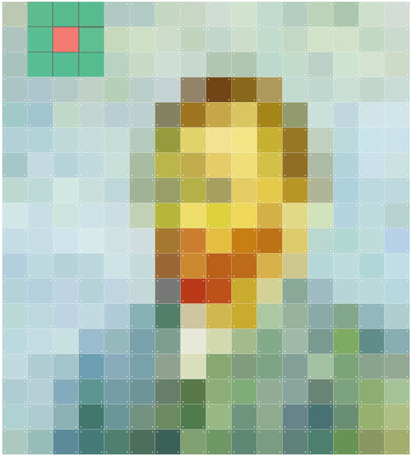


Graph Data

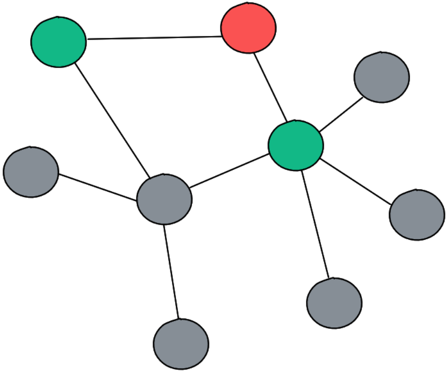


Problem Setup

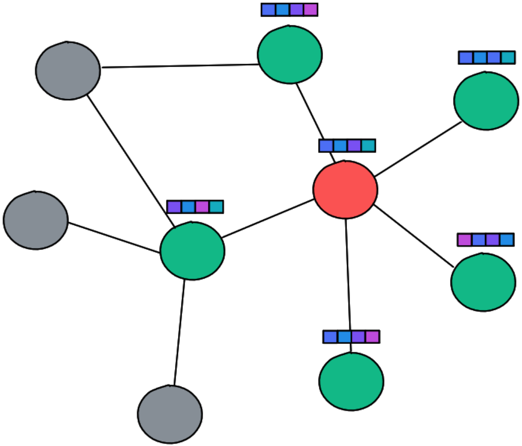
Grid Data



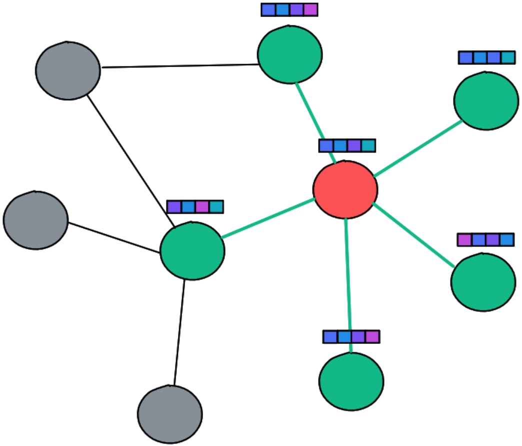
Graph Data



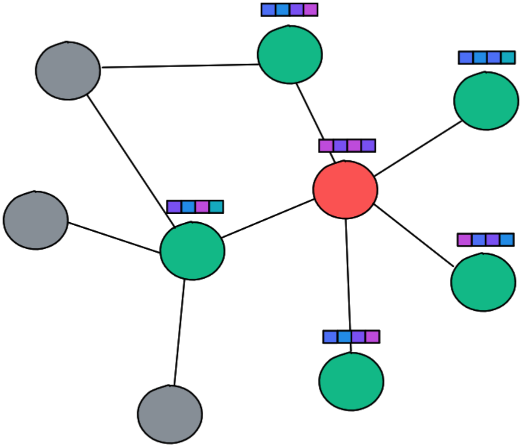
Graph Neural Networks



Graph Neural Networks



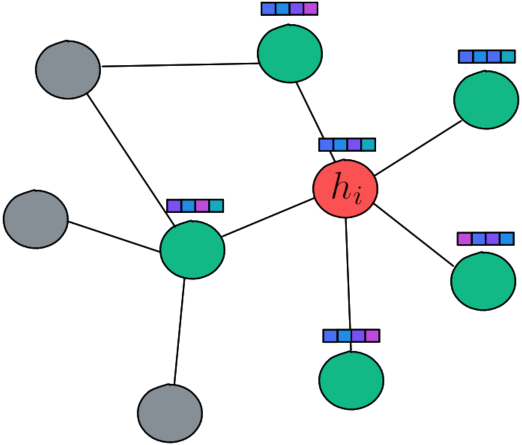
Graph Neural Networks



GAT Layer

$h_i \in \mathbb{R}^F,\ h'_i \in \mathbb{R}^{F'}$

feature weight matrix $\mathbf{W} \in \mathbb{R}^{F' \times F}$
shared attention mechanism
$a : \mathbb{R}^{F'} \times \mathbb{R}^{F'} \rightarrow \mathbb{R}$

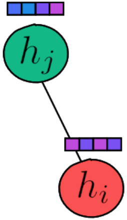


GAT Layer

$h_i \in \mathbb{R}^F$, $h_i' \in \mathbb{R}^{F'}$

feature weight matrix $\mathbf{W} \in \mathbb{R}^{F' \times F}$
shared attention mechanism
$a : \mathbb{R}^{F'} \times \mathbb{R}^{F'} \to \mathbb{R}$

attention coefficients
$e_{ij} = a(\mathbf{W}h_i, \mathbf{W}h_j)$



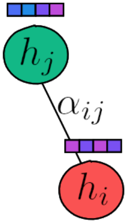
GAT Layer

$h_i \in \mathbb{R}^F$, $h'_i \in \mathbb{R}^{F'}$

feature weight matrix $\mathbf{W} \in \mathbb{R}^{F' \times F}$
shared attention mechanism
$a : \mathbb{R}^{F'} \times \mathbb{R}^{F'} \to \mathbb{R}$

attention coefficients
$e_{ij} = a(\mathbf{W}h_i, \mathbf{W}h_j)$

normalized attention coefficients
$\alpha_{ij} = \text{softmax}_j(e_{ij})$



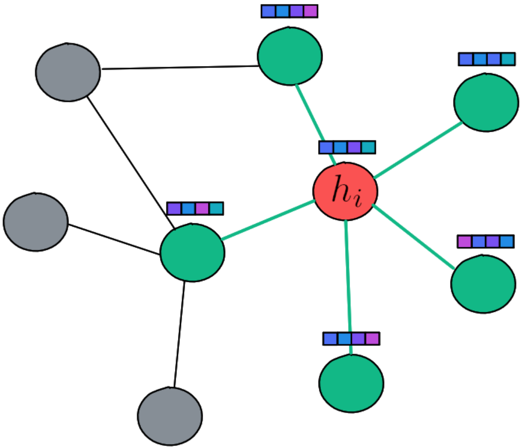
GAT Layer

$h_i \in \mathbb{R}^F$, $h_i' \in \mathbb{R}^{F'}$

feature weight matrix $\mathbf{W} \in \mathbb{R}^{F' \times F}$
shared attention mechanism
$a : \mathbb{R}^{F'} \times \mathbb{R}^{F'} \to \mathbb{R}$

attention coefficients
$e_{ij} = a(\mathbf{W}h_i, \mathbf{W}h_j)$

normalized attention coefficients
$\alpha_{ij} = \text{softmax}_j(e_{ij})$



GAT Layer

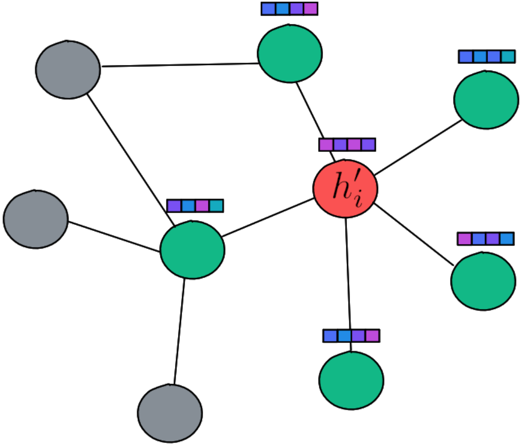
$h_i \in \mathbb{R}^F$, $h_i' \in \mathbb{R}^{F'}$

feature weight matrix $\mathbf{W} \in \mathbb{R}^{F' \times F}$
shared attention mechanism
$a : \mathbb{R}^{F'} \times \mathbb{R}^{F'} \to \mathbb{R}$

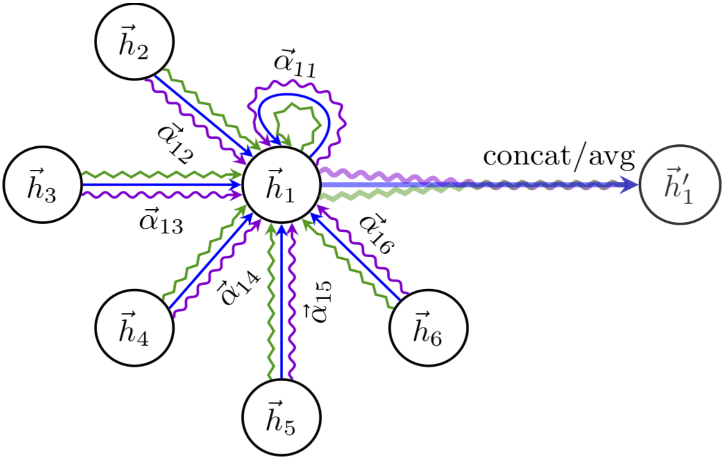
attention coefficients
$e_{ij} = a(\mathbf{W}h_i, \mathbf{W}h_j)$

normalized attention coefficients
$\alpha_{ij} = \text{softmax}_j(e_{ij})$

output with nonlinearity σ
$$h_i' = \sigma(\sum_{j \in \mathcal{N}_i} \alpha_{ij} \mathbf{W} h_j)$$



Multi-Head Attention



Veličković *et al.* 2018

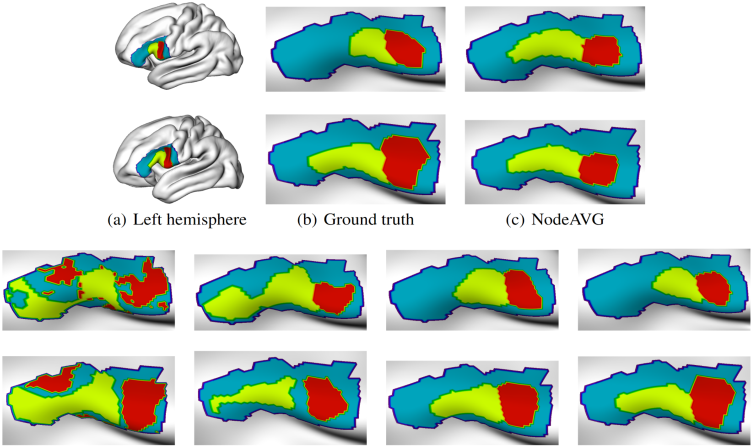
Improvements over previous methods

- shared attention mechanism allows usage for **inductive** learning tasks
- previous inductive methods did not include all neighbors
- attention weights lead to more interpretability

Results using GATs

- Classification of Brain Surface Areas (Cucurull *et al.*, 2018)
- Explainable Text Models (Rao *et al.*, 2021)

Brain Mesh Segmentation

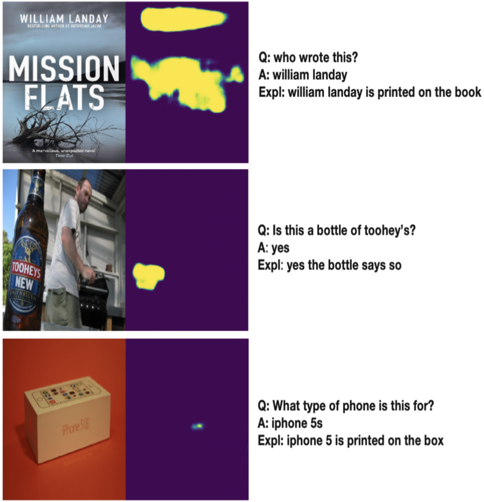


(a) Left hemisphere (b) Ground truth (c) NodeAVG

(d) NodeMLP (e) Jakobsen et al. [21] (f) GCN (g) GAT

Cucurull *et al.* 2018

Explainable Text Visual Question Answering Models



Rao *et al.* 2021

Explainable Text Visual Question Answering Models

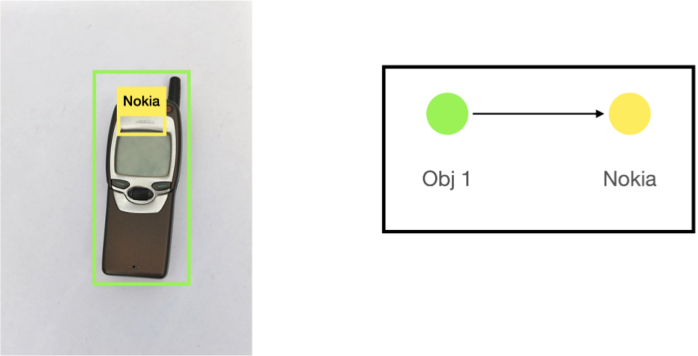


Figure 4: An example of how to build the graph

Rao *et al.* 2021

Explainable Text Visual Question Answering Models

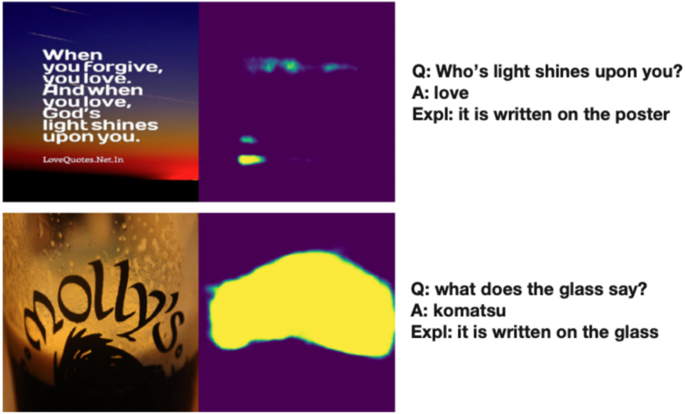


Figure 6: Examples where the MTXNet model fails.

Rao *et al.* 2021

Related Work and Limitations

- Benchmarking Graph Neural Networks (Dwivedi *et al.* 2020)
- Do we need deep graph neural networks? (Bronstein 2020)

Benchmarking GNNs

Cora and Citeseer are not large enough

Benchmarking GNNs

Table 6: Performance on the TU datasets with 10-fold cross validation (higher is better). Two runs of all the experiments using the same hyperparameters but different random seeds are shown separately to note the differences in ranking and variation for reproducibility. The top 3 performance scores are highlighted as **First**, **Second**, **Third**.

Dataset	Model	L	#Param	seed 1				seed 2			
				Test Acc.±s.d.	Train Acc.±s.d.	#Epoch	Epoch/Total	Test Acc.±s.d.	Train Acc.±s.d.	#Epoch	Epoch/Total
ENZYMES	MLP	4	101481	55.833±3.516	93.062±7.551	332.30	0.18s/0.17hr	53.833±4.717	87.854±10.765	327.80	0.19s/0.18hr
	GCN	4	103407	**65.833±4.610**	97.688±3.064	343.00	0.69s/0.67hr	64.833±7.089	93.042±4.982	334.30	0.74s/0.70hr
	GraphSage	4	105595	65.000±4.944	100.000±0.000	294.20	1.62s/1.34hr	**68.167±5.449**	100.000±0.000	287.30	1.76s/1.42hr
	MoNet	4	105307	63.000±8.090	95.229±5.864	333.70	0.53s/0.49hr	62.167±4.833	93.562±5.897	324.40	0.68s/0.62hr
	GAT	4	101274	**68.500±5.241**	100.000±0.000	299.30	0.70s/0.59hr	**68.500±4.622**	100.000±0.000	309.10	0.76s/0.66hr
	GatedGCN	4	103409	**65.667±4.899**	99.979±0.062	316.80	2.31s/2.05hr	**70.000±4.944**	99.979±0.062	313.20	2.63s/2.30hr
	GIN	4	104864	65.333±6.823	100.000±0.000	402.10	0.53s/0.61hr	67.667±5.831	100.000±0.000	404.90	0.60s/0.68hr
	RingGNN	2	103538	18.667±1.795	20.104±2.166	337.30	7.12s/6.71hr	45.333±4.522	56.792±6.081	497.50	8.05s/11.16hr
	3WLGNN	3	104658	61.000±6.799	98.875±1.571	381.80	9.22s/9.83hr	57.667±9.522	96.729±5.525	336.50	11.80s/11.09hr

Dwivedi *et al.* 2020

Benchmarking GNNs

different tasks favor different models

Benchmarking GNNs

Table 2: Benchmarking results for MP-GCNs and WL-GNNs across 7 medium-scale graph classification/regression and node/link prediction datasets. Results are averaged over 4 runs with 4 different seeds. **Red**: the best model, Violet: good models.

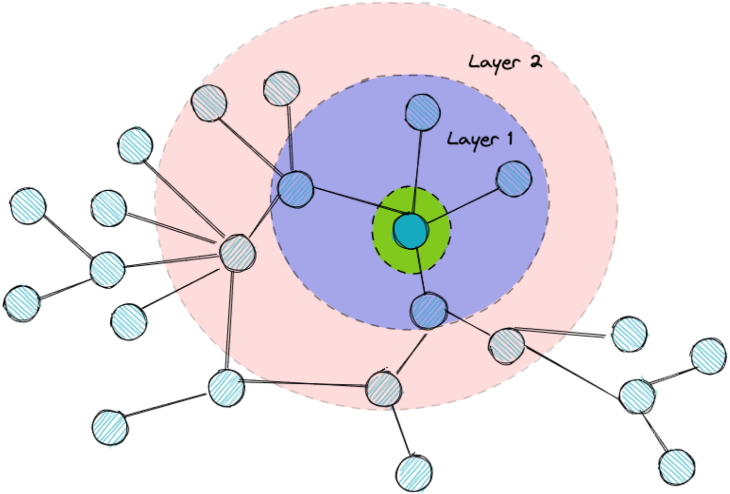
		NODE CLASSIFICATION									
		PATTERN					CLUSTER				
Model	L	**#Param**	**Test Acc.±s.d.**	**Train Acc.±s.d.**	**#Epoch**	**Epoch/Total**	**#Param**	**Test Acc.±s.d.**	**Train Acc.±s.d.**	**#Epoch**	**Epoch/Total**
MLP	4	105263	50.519±0.000	50.487±0.014	42.25	8.95s/0.11hr	106015	20.973±0.004	20.938±0.002	42.25	5.83s/0.07hr
GCN	4	100923	63.880±0.074	65.126±0.135	105.00	118.85s/3.51hr	101655	53.445±2.029	54.041±2.197	70.00	65.72s/1.30hr
	16	500823	71.892±0.334	78.409±1.592	81.50	492.19s/11.31hr	501687	68.498±0.976	71.729±2.212	79.75	270.28s/6.08hr
GraphSage	4	101739	50.516±0.001	50.473±0.014	43.75	93.41s/1.17hr	102187	50.454±0.145	54.374±0.203	64.00	53.56s/0.97hr
	16	502842	50.492±0.001	50.487±0.005	46.50	391.19s/5.19hr	503350	63.844±0.110	86.710±0.167	57.75	225.61s/3.70hr
MoNet	4	103775	85.482±0.037	85.569±0.044	89.75	35.71s/0.90hr	104227	58.064±0.131	58.454±0.183	76.25	24.29s/0.52hr
	16	511487	85.582±0.038	85.720±0.068	81.75	68.49s/1.58hr	511999	66.407±0.540	67.727±0.649	77.75	47.82s/1.05hr
GAT	4	109936	75.824±1.823	77.883±1.632	96.00	20.92s/0.57hr	110700	57.732±0.323	58.331±0.342	67.25	14.17s/0.27hr
	16	526990	78.271±0.186	90.212±0.476	53.50	50.33s/0.77hr	527874	70.587±0.447	76.074±1.362	73.50	35.94s/0.75hr
GatedGCN	4	104003	84.480±0.122	84.474±0.155	78.75	139.01s/3.09hr	104355	60.404±0.419	61.618±0.536	94.50	79.97s/2.13hr
	16	502223	85.568±0.088	86.007±0.123	65.25	644.71s/11.91hr	502615	73.840±0.326	87.880±0.908	60.00	400.07s/6.81hr
GatedGCN-PE	16	502457	**86.508±0.085**	86.801±0.133	65.75	647.94s/12.08hr	504253	**76.082±0.196**	88.919±0.720	57.75	399.66s/6.58hr

Dwivedi *et al.* 2020

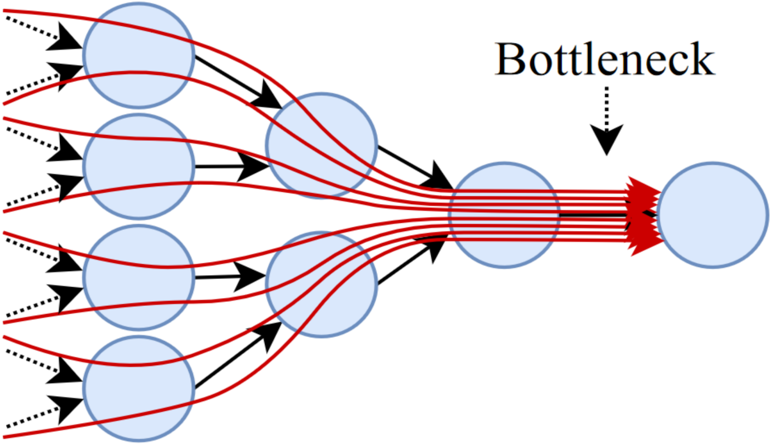
Going deep with GNNs

How far can we go with GATs?

Going deep with GNNs



Going deep with GNNs



Alon & Yahav 2021

References

Veličković *et al.* 2018
Graph Attention Networks
arXiv 1710.10903
github.com/PetarV-/GAT

Cucurull *et al.* 2018
Convolutional neural networks for mesh-based parcellation of the cerebral cortex
MIDL 2018 Conference Submission
https://openreview.net/forum?id=rkKvBAiiz

Rao *et al.* 2021
A First Look: Towards Explainable TextVQA Models via Visual and Textual Explanations
arXiv 2105.02626

References

Dwivedi *et al.* 2020
Benchmarking Graph Neural Networks
arXiv 2003.00982

Bronstein 2020
Do we need deep graph neural networks?
https://towardsdatascience.com/do-we-need-deep-graph-neural-networks-be62d3ec5c59

Alon & Yahav 2021
On the Bottleneck of Graph Neural Networks and its Practical Implications
arXiv 2006.05205