How Attentive are Graph Attention Networks? not that much...



Shaked Brody



Uri Alon Technion Language Technologies Institute **Carnegie Mellon University**



Eran Yahav Technion

https://arxiv.org > stat . .

[1710.10903] Graph Attention Networks - arXiv

by P Veličković · 2017 · Cited by 5222 — Abstract: We present graph attention networks (GATs), novel neural network architectures that operate on graph-structured data, ...

we introduce an attention-based architecture to perform node classification of graph-structured data. The idea is to compute the hidden representations of each node in the graph, by attending over its neighbors, following a *self-attention* strategy. The attention architecture

But what kind of attention?

Attention



The ability of different queries to learn to "focus" differently on a set of keys

[Bahdanau et al., ICLR 2015]

Graph Attention Networks (GAT) [Veličković et al., 2018]

	<i>k</i> 0	k1	k2	k3	k4	<i>k</i> 5	<i>k</i> 6	k7	k8	k9
q0 -	0.08	0.10	0.10	0.07	0.08	0.08	0.11	0.09	0.20	0.08
q1 -	0.05	0.10	0.10	0.04	0.04	0.04	0.13	0.06	0.38	0.04
q2 -	0.05	0.10	0.10	0.04	0.05	0.05	0.13	0.06	0.38	0.05
q3 -	0.08	0.10	0.10	0.07	0.08	0.08	0.10	0.09	0.24	0.08
q4 -	0.08	0.09	0.09	0.07	0.07	0.07	0.10	0.08	0.27	0.07
q5 -	0.09	0.11	0.11	0.08	0.09	0.08	0.11	0.10	0.16	0.09
q6 -	0.04	0.10	0.11	0.03	0.04	0.04	0.14	0.06	0.40	0.04
q7 -	0.07	0.09	0.09	0.06	0.07	0.07	0.10	0.08	0.29	0.07
q8 -	0.04	0.11	0.11	0.02	0.04	0.03	0.14	0.07	0.41	0.04
q9 -	0.07	0.09	0.09	0.06	0.07	0.07	0.11	0.08	0.30	0.07

Static Attention

GAT uses an Addition of Two Dot Products $e(h_i, h_j) = LeakyRe$



 $e(h_i, h_j) = LeakyReI$ $\mathbb{R}^1 \to \mathbb{R}$

$$eLU(a^{\mathsf{T}}[Wh_i \mid \mid Wh_j])$$



$$eLU(a_1^{\mathsf{T}} \cdot Wh_i + a_2^{\mathsf{T}} \cdot Wh_j)$$

$$\mathbb{R}^1 \in \mathbb{R}^1 \quad \in \mathbb{R}^1$$

GAT Attends to the Same Key Regardless of Query

 $e(h_i, h_j) = LeakyReL$

	<i>k</i> 0	<i>k</i> 1	k2	k3	<i>k</i> 4	<i>k</i> 5	<i>k</i> 6	k7	k8	k9
q0 -	0.08	0.10	0.10	0.07	0.08	0.08	0.11	0.0	0.20	0.08
q1 -	0.05	0.10	0.10	0.04	0.04	0.04	0.13	0.05	0.38	0.04
q2 -	0.05	0.10	0.10	0.04	0.05	0.05	0.13	0.05	0.38	0.05
q3 -	0.08	0.10	0.10	0.07	0.08	0.08	0.10	0.0	0.24	0.08
q4 -	0.08	0.09	0.09	0.07	0.07	0.07	0.10	0.03	0.27	0.07
q5 -	0.09	0.11	0.11	0.08	0.09	0.08	0.11	0.1	0.16	0.09
q6 -	0.04	0.10	0.11	0.03	0.04	0.04	0.14	0.0	0.40	0.04
q7 -	0.07	0.09	0.09	0.06	0.07	0.07	0.10	0.03	0.29	0.07
q8 -	0.04	0.11	0.11	0.02	0.04	0.03	0.14	0.07	0.41	0.04
q9 -	0.07	0.09	0.09	0.06	0.07	0.07	0.11	0.03	0.30	0.07
		C+	~ +		V † 1		. +:		$\overline{}$	

Static Attention

$$U(a_1^{\mathsf{T}} \cdot Wh_i + a_2^{\mathsf{T}} \cdot Wh_j)$$

 S_j

 $S_8 \ge S_6 \ge S_2 \ge \cdots$

GATv2: Fixing Graph Attention Mechanism

GAT, Veličković et al., 2018:

GATv2, this work:

GAT										
	<i>k</i> 0	<i>k</i> 1	k2	k3	<i>k</i> 4	<i>k</i> 5	<i>k</i> 6	k7	k8	k9
q0 -	0.08	0.10	0.10	0.07	0.08	0.08	0.11	0.09	0.20	0.08
q1 -	0.05	0.10	0.10	0.04	0.04	0.04	0.13	0.0	0.38	0.04
q2 -	0.05	0.10	0.10	0.04	0.05	0.05	0.13	0.0	0.38	0.05
q3 -	0.08	0.10	0.10	0.07	0.08	0.08	0.10	0.09	0.24	0.08
q4 -	0.08	0.09	0.09	0.07	0.07	0.07	0.10	0.0	0.27).07
q5 -	0.09	0.11	0.11	0.08	0.09	0.08	0.11	0.1(0.16	0.09
q6 -	0.04	0.10	0.11	0.03	0.04	0.04	0.14	0.0	0.40	0.04
q7 -	0.07	0.09	0.09	0.06	0.07	0.07	0.10	0.08	0.29).07
q8 -	0.04	0.11	0.11	0.02	0.04	0.03	0.14	0.0	0.41	0.04
q9 -	0.07	0.09	0.09	0.06	0.07	0.07	0.11	0.08	0.30	0.07

Static Attention

$e(h_i, h_j) = LeakyReLU(a^{\mathsf{T}}[Wh_i || Wh_j])$

 $e(h_i, h_j) = a^{\mathsf{T}} LeakyReLU(W \cdot [h_i || h_j])$

GATv2

	<i>k</i> 0	<i>k</i> 1	k2	<i>k</i> 3	<i>k</i> 4	<i>k</i> 5	<i>k</i> 6	k7	<i>k</i> 8	<i>k</i> 9
q0 -	0.95	0.00	0.00	0.01	0.01	0.00	0.00	0.02	0.01	0.00
q1 -	9.01	0.92	0.01	0.01	0.01	0.00	0.01	0.01	0.00	0.02
q2 -	0.00	2,00	0.95	0.00	0.00	0.01	0.02	0.01	0.00	0.00
q3 -	0.01	0.01	8 00	0.94	0.90	0.01	0.00	0.00	0.02	0.01
q4 -	0.00	0.00	0.00	00 0	0.96	0.20	0.00	0.01	0.01	0.00
q5 -	0.00	0.01	0.01	0.01	0.01	0.89	0.81	0.01	0.04	0.02
q6 -	0.00	0.01	0.04	0.00	0.01	0.01	0.86	0.02	0.01	0.03
q7 -	0.04	0.02	0.01	0.01	0.03	0.01	0.00	0.87	0.00	0.01
q8 -	0.01	0.00	0.01	0.01	0.01	0.01	0.01	0.00	0.94	0.00
q9 -	0.01	0.02	0.01	0.01	0.01	0.01	0.01	0.00	0.90	0.93

Dynamic Attention

Static Attention

For any sets of:





There is always a key that gets the most attention, regardless of the query

Dynamic Attention

For any set of queries, keys, and any desired mapping between them:







There exist learned parameters that "implement" this mapping

Experimental Results

- **GATv2** always outperformed **GAT** in 12 benchmarks of node-link- and graph-prediction **GATv2** is more robust to noisy edges (which did not exist in the original graph)



Summary

- Define static attention vs. dynamic attention
- **GAT** computes static attention
- **GATv2**: a simple modification that is strictly more expressive than **GAT**
 - More accurate across 12 benchmarks and more robust to noise
- Use **GATv2** instead of **GAT** whenever possible
- **GATv2** is available on:
 - PyTorch Geometric:
 - DGL:
 - **TensorFlow GNN:**

shakedbr@cs.technion.ac.il

http://shakedbr.cswp.cs.technion.ac.il

from torch_geometric.nn **import** GATv2Conv **from** dgl.nn.pytorch **import** GATv2Conv **from** tensorflow_gnn.keras.layers **import** GATv2