GNNs and Graph Processing

Oliver Hope
What is a GNN?

- A type of neural network which directly operates on the Graph structure
- Learns an embedding for each node based on features and neighbours
- Often runs using an iterative update approach
- Libraries exist to help implementation such as Graph Nets[1]
Ideas

1: Let’s use it to run graph algorithms

- Some algorithms take a long time to run
- Sometimes we only need so much accuracy
- Sacrifice accuracy for runtime but still get good results?[2]
- Maybe we don’t know the algorithm but do have data

2: Let’s run it on a graph processing backend

- Execution requires message passing between neighbours
- This is in an iterated process
- Would a graph processing backend be more efficient than current libraries?
- E.g. Graph Nets is built on TensorFlow but how well does it use the dataflow?
Plan

- Graph Nets run over TensorFlow
- TensorFlow inspired by Naiad, but more limited
- Will benchmark simple algorithm over Graph Nets API (i.e., break open the codebase, bypass the neural parts) and Naiad
- If time, incorporate Naiad ideas in Graph Nets and assess impact against “vanilla” implementation
- This (may) open up many possibilities for further exploration.
Questions?