from torchviz import make\_dot

from graphviz import Digraph

x = torch.zeros((1, n\_features))

edge\_index = torch.zeros((2, 2), dtype=torch.long)

model = student\_model

out = model(x, edge\_index)

dot = make\_dot(out, params=dict(model.named\_parameters()))

# Add legend

dot.node('Input', label='Input\n{}'.format(x.shape), shape='rectangle')

dot.node('Output', label='Output\n{}'.format(out.shape), shape='rectangle')

dot.node('Parameters', label='Parameters:', shape='rectangle')

for name, param in model.named\_parameters():

dot.node(name, label='Parameter\n{}'.format(param.shape), shape='rectangle')

# Set edge direction and add legend

dot.graph\_attr.update(directed='True')

dot.attr('node', shape='rectangle', style='filled', fillcolor='#d3d3d3')

dot.attr('edge', arrowsize='0.7')

dot.attr('graph', rankdir='TB')

dot.render('model\_architecture\_GAT2', format='pdf')