

The graph neural network (GNN) has become a popular research area for its state-of-the-art performance in many graph analysis tasks. Recently, various graph neural network libraries have emerged. They make the development of GNNs convenient, but their performance on large datasets is not satisfying. In this work, we analyze the performance bottleneck in training GNN with GPUs empirically. A GNN layer can be decomposed into two parts: the vertex and the edge calculation parts. According to their computational complexity, we select four representative GNNs (GCN, GGNN, GAT, GaAN) for evaluation. We breakdown their training time and memory usage, evaluate the effects of hyper-parameters, and assess the efficiency of the sampling techniques. The experimental evaluation indicates that the edge-related calculation is the performance bottleneck for most GNNs, dominating the training time and memory usage. Future optimization can focus on it. The sampling techniques are essential for training big graphs on GPUs, but their current implementations still have room for improvement.