Karlsruher Institut für Technologie
Fakultät für Informatik

Lehrstuhl Algorithm Engineering
Prof. Dr. Peter Sanders

Bachelor's or Master's Thesis

Machine Learning for Graph Partitioning

Description

Graph partitioning is an NP-hard optimization problem, where the task is to divide the nodes of a graph into k roughly equal size parts while minimizing the number of edges running between different parts. This problem has numerous applications such as optimizing communication volume under load balance in distributed computing.

In this project, we want to investigate how machine learning can improve state-of-the-art graph partitioning heuristics. Most graph partitioning approaches follow the three-phase multilevel scheme of *coarsening, initial partitioning and refinement*. In the coarsening phase, one builds a hierarchy of increasingly smaller graphs by contracting node clusters, which aims to approximately preserve the graph structure. As such, the initial partition computed on the smallest graph is already a decent solution. In the refinement phase, the contractions are undone in reverse order, with local search improving the partition on each level. Moving a node on coarse levels corresponds to moving a cluster of nodes on fine levels. Thus, multilevel algorithms perform global optimization via local search.

The initial step involves identifying relevant graph features that can aid in predicting regions suitable for local search. These features might include node degrees, neighborhood information, and graph structure metrics. Then, feature engineering techniques shall be employed to extract meaningful representations from the graph data.
To enable prediction of promising search regions, a machine learning model shall be trained. Reinforcement learning is a promising approach, where the agent learns to make decisions based on observed rewards. To create a sufficiently large training set, we could use synthetic graph data, such as RMAT graphs. The model will learn to predict nodes or clusters that are likely to yield improvements during local search.

Goals

The outcome of this project is an implementation of a machine learning model to improve graph partitioning heuristics. The implementation should be carried out in one of our graph partitioning frameworks, which will need to combine graph partitioning code in C++ and machine learning code in Python. Subsequently, the approaches shall be evaluated in terms of running time and partition quality.

Requirements

- Interest in machine learning, and beginner-level experience in algorithms and data structures
- Highly self-motivated to learn new concepts
- Good programming skills in Python and intermediate knowledge of C++
- Prior knowledge of machine learning is desirable but not required

Note

This thesis project is jointly supervised with Dr. Deepak Ajwani from University College Dublin.

Lars Gottesbüren, Nikolai Maas, Daniel Seemaier
{lars.gottesbueren, nikolai.maas, daniel.seemaier}@kit.edu