Memory-Efficient Neural Networks

Bachelor’s Thesis or Seminar/Master’s Project

Today's neural networks are split into deep stacks of layers. While computing the networks output, all intermediate outputs of layers are stored because they are needed during training to compute parameter changes (backpropagation). The use of big models is limited by GPU-memory. We want to reduce the memory-demands during training in this thesis.

Objective:
You will get familiar with how training of deep models works. The above figure shows an abstract model of allocated resources in a neural network. The model consists of layers (L), the computed intermediate results (A) and the errors (E) needed for training. The red circles make up for most of the memory consumption in state-of-the-art networks. The aim is to discard most intermediate results (A) during forward computation and only store a few to allow recomputation of the discarded results later on when needed during the backwards path. The implementation will be done in Tensorflow [1]. Similar approaches are already available in other frameworks [2].

Qualifications:
- Student of Information and Computer Engineering, Computer Science or Software Engineering and Management
- Basic knowledge of neural networks
- Programming experience in Python and C++
- Passion for coding

Contact:
Aleksander Colovic	Thomas Pock
aleksander.colovic@icg.tugraz.at	pock@icg.tugraz.at

References