note of 'MEPAD: A Memory-Efficient Parallelized Direct Convolution Algorithm for Deep Neural Networks'

zxp

December 28, 2024

1 content

This is a paper from 24 years. The main work is directly convolving the optimization strategy on different memory architectures (scene high-performance machines or custom accelerators). The optimization used in the paper is also very conventional SIMD and block. Paper usage is also focused on memory performance, number of reads and other memory usage metrics.

1.1 BACKGROUND

The paper say that the most appropriate layout depends on the chosen convolutional implementation and the target computing architecture. The paper uses two different ways of storing tensors (data layout), row-first and channel first. Like many papers, this paper also says that since the output of most convolutional layers is the input of another convolutional layer, the output and input activation matrices should have the same storage layout to avoid expensive data reorganization. The paper mentions some other optimization directions, which use heuristic methods to search extensively in a large space for optima. The paper argues that their work, based on minimizing the number of accesses to the last level of the memory hierarchy per operation, leverages existing knowledge of the computing architecture to select the most promising directions and is batter.

1.2 Reference Architecture

The accelerated architecture consists of multiple single Instruction Multiple Data (SIMD) processing units (pes) connected by high-speed and high-bandwidth interconnects. It supports FMA and has plenty of registers. The paper's work

is on two memory systems, the first two-level memory hierarchy where PEs are connected to shared on-chip scratchpad memory (SPM) that behaves like off-chip memory (DDR3) and a cache between PEs (which should be more common memory systems). The second Direct Connection High Bandwidth Memory (HBM)

1.3 Convolution

The paper points out the characteristics of convolution, there are a lot of reusable data can be reused in different pes, block can make good use of the reused data, each pe computing different block parallel way is more efficient. We introduce different convolutions, summarize some work done by different authors, mention some papers I haven't seen, MEC and kn2row/kn2col/kn2row-l/kn2col-l. The paper also implements im2row-d (not described))

1.4 MEPAD

The goal of the paper is to minimize the access to the memory subsystem and the system energy consumption by maximizing the time and space sharing of data. To this end, we need to block (in fact, there is no specific strategy, did not say how to split, just say to split, how to calculate after the dismantling...) . Then, the output tensor is divided into different parts for calculation by different pes. One is that each pe has the same filter, and the output tensor is divided into rows and columns. The other is that the output tensor is divided into channels (split filter).

1.5 Experimental

The point is to analyze the number of memory reads, and the feature is to estimate the energy required by the convolution by the number of reads and the cost of reads (the consumption of reads from different levels varies). The paper says that his method requires fewer memory accesses (especially memory far from the cpu). The paper boils down to their parallel design and read memory with finer granularity

2 Feelings

The paper is written from the perspective of memory reads. The section summarizing other convolutional methods has some reference value The paper is about Time series analysis. it use convolution, not how to implement convolution, no reference