Extended Abstract

Review and Research: Journal of Engineering And Technology

2021

Vol. 10 Issue 2

Design productivity, compilation and acceleration for data analytic applications

Deming Chen
University of Illinois at Urbana-Champaign,USA

Abstract:

Deep Neural Networks (DNNs) are computation intensive. Without efficient hardware implementations of DNNs, many promising AI applications will not be practically realizable. In this talk, we will analyze several challenges facing the AI community for mapping DNNs to hardware accelerators. Especially, we will evaluate FPGA's potential role in accelerating DNNs for both the cloud and edge devices. Although FPGAs can provide desirable customized hardware solutions, they are difficult to program and optimize. We will present a series of effective design techniques for implementing DNNs on FPGAs with high performance and energy efficiency. These include automated hardware/software co-design, the use of configurable DNN IPs, and resource allocation across DNN layers, smart pipeline scheduling, Winograd and FFT techniques and DNN reduction and re-training. We showcase several design solutions including Long-term Recurrent Convolution Network (LRCN) for video captioning, Inception module (Google Net) for face recognition, as well as Long Short-Term Memory (LSTM) for sound recognition. We will also present some of our recent work on developing new DNN models and data structures for achieving higher accuracy for several interesting applications such as crowd counting, genomics and music synthesis. Fundamental data analytics tasks are often simple – many useful and actionable insights can be garnered by simply filtering, grouping, and summarizing data. However the sheer volume of data to be analyzed, demands of a multi-user operating environment, and limitations of general purpose processors make it challenging to perform these operations efficiently at scale. This thesis presents two techniques that address these challenges to improve the response time of data analytics tasks: (1) newly emerging programmable network processors can perform data analytics tasks at terabits per second. However, existing data analytics systems, like Apache Spark, cannot readily use network processors because network processors are very limited and cannot execute the tasks generated by existing analytics systems. Using network processors for analytics requires re-designing how existing systems compile and execute data processing tasks. This thesis introduces Jump gate, a system that enables existing data processing systems to execute relational queries using network processors. Jump gate compiles client requests to a novel execution model that coordinates execution on heterogeneous network processors. Jump gate can already improve performance by 1.12-3× on industry standard benchmarks, and paves the way for adopting network processors for data analytics tasks. (2) Analytics systems often process similar queries, either submitted by the same or different users. Cross program memorization (CPM) is a technique to re-use results of prior computations across programs and users. However, CPM is often not enabled because prior implementations have high overhead and are unable to reuse the output of user-defined functions (UDFs). This thesis presents Key Chain, a CPM implementation that identifies equivalent UDFs and has low overhead so CPM can be always be enabled.

This work is partly presented at 7th International Conference on Big Data Analytics & Data Mining | September 26-27, 2018 | Chicago, USA