DeepX: A Software Accelerator for Squeezing Deep Learning onto Wearables, Phones and Things

Nicholas D. Lane[‡], Sourav Bhattacharya[‡]
Petko Georgiev[†], Claudio Forlivesi[‡], Fahim Kawsar[‡]

[‡]Bell Labs, [†]University of Cambridge

1. INTRODUCTION

Breakthroughs from the field of deep learning are radically changing how sensor data are interpreted to extract the high-level information needed by mobile apps [1]. It is critical that the gains in inference accuracy that deep models afford become embedded in future generations of mobile apps. Unfortunately this is not yet happening – even though mobile apps present some of the most challenging examples of noisy sensor data to which inference models are applied. In this demonstration we present the *DeepX* prototype, a software accelerator for both Deep Neural Networks and Convolutional Neural Networks. DeepX significantly lowers the device resources (viz. memory, computation, energy) required by deep learning are a severe bottleneck to mobile adoption.

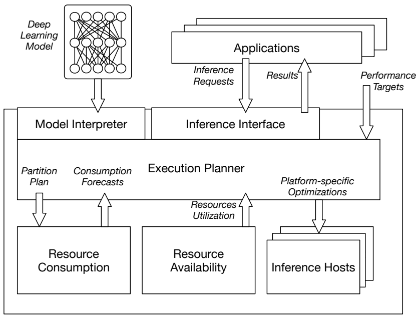


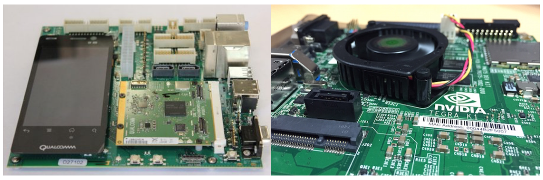
Figure 1: DeepX Proof-of-Concept System

2. DESIGN AND OPERATION

DeepX aims to radically reduce mobile resource use (viz. memory, computation and energy), in addition to the execution time, of performing inference with large-scale deep learning models by exploiting a mix of network-based computation and heterogeneous local processors. Towards this goal, we propose two novel techniques:

- **Runtime Layer Compaction (RLC):** A building block to optimizing mobile resource usage for deep learning is an ability to shape and control them. But existing approaches, such as those of model compression, focus on the training phase of deep learning models, rather than the inference. RLC provides runtime control of the memory and computation (along with energy as a side-effect) consumed during the inference phase by extending model compression principles.
- **Deep Architecture Decomposition (DAD):** A typical deep model is comprised of an architecture of many layers and thousands of units. DAD efficiently identifies unit-blocks of this architecture and creates a "decomposition plan" that allocates blocks to local and remote processors; such plans maximize resource utilization and seek to satisfy user performance goals.

To demonstrate and evaluate the algorithms of RLC and DAD, and the end-to-end operation of DeepX, we develop a proof-of-concept system shown in Figure 1. Any already trained DNN or CNN model can be provided to DeepX. Model specifications come from developers who then incorporate the use of DeepX into the logic of a mobile or wearable app. Requests to perform an inference using an earlier provided model are made via an API. A developer then includes an API call within a mobile app. Each time an inference is requested DeepX determines a new plan for execution. This enables the execution plan to be optimized for the current local device resource conditions. Supporting *efficient* inference operations across many processor types force the use of many host implementations that include platform specific optimizations. Two prototype versions of DeepX have been built (shown in Figure 2), one for the Qualcomm Snapdragon 800, and the other targets the Nvidia Tegra K1.



(a) Snapdragon 800 **(b)** Tegra K1

Figure 2: Developer Boards for SoCs used for DeepX Prototype

3. PROTOTYPE PERFORMANCE

We briefly highlight representative performance benefits of DeepX under two large-scale deep models that were originally conceived for the cloud. The first model, *AlexNet* [3], performs object recognition and supports more than 1,000 object classes. The second model, *CD-Seide* [2], is used for speech recognition and understands more than 31K words. We find when running on the Tegra K1, DeepX improves the energy efficiency of AlexNet by 22.1×, 1.8× and 1.3× compared to benchmarks that use cloud computation, and GPU- or CPU-only solutions, respectively. Under CD-Seide, these numbers are: 7.8×, 1.4× and 29.7×. DeepX on the Snapdragon 800, for the same models, has similar energy benefits: AlexNet, 11.2× (cloud) and 2.1× (CPU); and CD-Seide, 8.2× (cloud) and 2.8× (CPU). DeepX also benefits memory and execution time bottlenecks, with execution tightly coupled to energy gains and reductions of model memory footprint of 2.5× typical for AlexNet and CD-Seide.

4. REFERENCES

[1] Y. Bengio, et al., "Deep learning," MIT Press 2015

[2] F. Seide, et al., "Conversational speech transcription using context-dependent deep neural networks." *Interspeech '11*

[3] A. Krizhevsky, et al., "Imagenet classification with deep convolutional neural networks," *NIPS '12*