Google Neural Network Models for Edge Devices: Analyzing and Mitigating Machine Learning Inference Bottlenecks

Computer Architecture, Lecture 23a
Fall 2022

Amirali Boroumand  Saugata Ghose  Berkin Akin
Ravi Narayanaswami  Geraldo F. Oliveira  Xiaoyu Ma
Eric Shiu  Onur Mutlu

PACT 2021

SAFARI

Carnegie Mellon  University of Illinois Urbana-Champaign  Google  ETH Zürich
Executive Summary

Context: We extensively analyze a state-of-the-art edge ML accelerator (Google Edge TPU) using 24 Google edge models
- Wide range of models (CNNs, LSTMs, Transducers, RCNNs)

Problem: The Edge TPU accelerator suffers from three challenges:
- It operates significantly below its peak throughput
- It operates significantly below its theoretical energy efficiency
- It inefficiently handles memory accesses

Key Insight: These shortcomings arise from the monolithic design of the Edge TPU accelerator
- The Edge TPU accelerator design does not account for layer heterogeneity

Key Mechanism: A new framework called Mensa
- Mensa consists of heterogeneous accelerators whose dataflow and hardware are specialized for specific families of layers

Key Results: We design a version of Mensa for Google edge ML models
- Mensa improves performance and energy by 3.0X and 3.1X
- Mensa reduces cost and improves area efficiency
Outline

1. Introduction
2. Edge TPU and Model Characterization
3. Mensa Framework
4. Mensa-G: Mensa for Google Edge Models
5. Evaluation
6. Conclusion
| Outline |
|---------|
| **1** Introduction |
| **2** Edge TPU and Model Characterization |
| **3** Mensa Framework |
| **4** Mensa-G: Mensa for Google Edge Models |
| **5** Evaluation |
| **6** Conclusion |
Why ML on Edge Devices?

Significant interest in pushing ML inference computation directly to edge devices

Privacy
Connectivity
Latency
Bandwidth
Why Specialized ML Accelerator?

Edge devices have limited battery and computation budget

- **Limited Power Budget**
- **Limited Computational Resources**

Specialized accelerators can significantly improve inference latency and energy consumption

- Apple Neural Engine (A12)
- Google Edge TPU
Myriad of Edge Neural Network Models

Challenge: edge ML accelerators have to execute inference efficiently across a wide variety of NN models
Edge TPU: Baseline Accelerator

- **ML Model**
- **DRAM**
- **Input Activation**
- **Parameter**
- **Output Activation**
- **Dataflow**

**PE Array**
- 64x64 array
- 2TFLOP/s
- 4MB on-chip buffer

**Mensa Framework**

**Conclusion**
Google Edge NN Models

We analyze inference execution using 24 edge NN models

Speech Recognition

Face Detection

Google Edge TPU

Image Captioning

Language Translation

6 RNN Transducers

13 CNN

2 LSTMs

3 RCNN

Introduction

TPU and Model Characterization

Mensa Framework

Mensa-G

Evaluation

Conclusion
Major Edge TPU Challenges

We find that the accelerator suffers from **three major challenges:**

1. Operates significantly below its peak throughput

2. Operates significantly below its peak energy efficiency

3. Handles memory accesses inefficiently
We find that the accelerator operates significantly below its peak throughput across all models.
Low Energy Efficiency

The accelerator operates far below its upper bound energy efficiency

Best CNN model: 50.7% of upper bound energy efficiency

LSTMs and Transducers: 33.1% of upper bound energy efficiency

Peak = 1.42 TFLOP/J
46% and 31% of total energy goes to off-chip parameter traffic and distributing parameters across PE array.
We find that the accelerator suffers from three major challenges:

1. Operates significantly below its peak throughput
2. Operates significantly below its peak energy efficiency
3. Handles memory accesses inefficiently

Question: Where do these challenges come from?
Model Analysis:
Let’s Take a Deeper Look Into the Google Edge NN Models
Insight 1: there is significant variation in terms of layer characteristics across the models.
Diversity Within the Models

**Insight 2**: even **within** each model, layers exhibit significant variation in terms of layer characteristics

For example, our analysis of edge CNN models shows:

Variation in **MAC intensity**: up to **200x** across layers

Variation in **FLOP/Byte**: up to **244x** across layers
The key components of Google Edge TPU are completely oblivious to layer heterogeneity.

Edge accelerators typically take a monolithic approach: equip the accelerator with an over-provisioned PE array and on-chip buffer, a rigid dataflow, and fixed off-chip bandwidth.

While this approach might work for a specific group of layers, it fails to efficiently execute inference across a wide variety of edge models.
Goal: design an edge accelerator that can efficiently run inference across a wide range of different models and layers

Instead of running the entire NN model on a monolithic accelerator:

Mensa: a new acceleration framework for edge NN inference
The goal of Mensa’s software runtime scheduler is to identify which accelerator each layer in an NN model should run on.

Generated once during initial setup of a system.

- Each of the accelerators caters to a specific family of layers.
- Layers tend to group together into a small number of families.
The goal of Mensa’s software runtime scheduler is to identify which accelerator each layer in an NN model should run on. Generated once during initial setup.

Layers tend to group together into a small number of families. Each of the accelerators caters to a specific family of layers.

Google Neural Network Models for Edge Devices: Analyzing and Mitigating Machine Learning Inference Bottlenecks

Amirali Boroumand†‡
Geraldo F. Oliveira* 
Saugata Ghose‡
Xiaoyu Ma§
Berkin Akin§
Eric Shiu§
Ravi Narayanaswami§
Onur Mutlu*†

†Carnegie Mellon Univ.  ‡Stanford Univ.  ‡Univ. of Illinois Urbana-Champaign  §Google  *ETH Zürich
Identifying Layer Families

Key observation: the majority of layers group into a small number of layer families

Families 1 & 2: low parameter footprint, high data reuse and MAC intensity → compute-centric layers

Families 3, 4 & 5: high parameter footprint, low data reuse and MAC intensity → data-centric layers
Mensa-G: Mensa for Google Edge Models

Based on key characteristics of families, we design three accelerators to efficiently execute inference across our Google NN models.
Mensa-G: Mensa for Google Edge Models

Based on key characteristics of families, we design three accelerators to efficiently execute inference across our Google NN models.

**Pascal**
- 32x32 PE Array
- DRAM (32 GB/s)
- Families 1&2 → compute-centric layers
  - 32x32 PE Array → 2 TFLOP/s
  - 256KB Act. Buffer → 8x Reduction
  - 128KB Param. Buffer → 32x Reduction
  - On-chip accelerator

**Pavlov**
- 8x8 PE Array
- DRAM (256 GB/s)

**Jacquard**
- 16x16 PE Array
- DRAM (256 GB/s)

SAFARI
- Introduction
- TPU and Model Characterization
- Mensa Framework
- Mensa-G
- Evaluation
- Conclusion

28
Based on **key characteristics** of families, we design **three accelerators** to efficiently execute inference across our Google NN models.

**Pascal**
- 32x32 PE Array → 2 TFLOP/s
- 256KB Act. Buffer → 8x Reduction
- 128KB Param. Buffer → 32x Reduction
- On-chip accelerator

**Pavlov**
- 8x8 PE Array → 128 GFLOP/s
- 128KB Act. Buffer → 16x Reduction
- No Param. Buffer → 4MB in Baseline
- Near-data accelerator

**Jacquard**
- 16x16 PE Array
- No specific details provided in the diagram

**Families 1&2** → compute-centric layers
- 32x32 PE Array
- 2 TFLOP/s
- 256KB Act. Buffer → 8x Reduction
- 128KB Param. Buffer → 32x Reduction
- On-chip accelerator

**Family 3** → LSTM data-centric layers
- 8x8 PE Array
- 128 GFLOP/s
- 128KB Act. Buffer → 16x Reduction
- No Param. Buffer → 4MB in Baseline
- Near-data accelerator
Based on **key characteristics** of families, we design **three accelerators** to efficiently execute inference across our Google NN models.

**Pascal**
- **Families 1&2** → compute-centric layers
  - 32x32 PE Array → 2 TFLOP/s
  - 256KB Act. Buffer → 8x Reduction
  - 128KB Param. Buffer → 32x Reduction
  - On-chip accelerator

**Pavlov**
- **Family 3** → LSTM data-centric layers
  - 8x8 PE Array → 128 GFLOP/s
  - 128KB Act. Buffer → 16x Reduction
  - No Param. Buffer → 4MB in Baseline
  - Near-data accelerator

**Jacquard**
- **Families 4&5** → non-LSTM data-centric layers
  - 16x16 PE Array → 256 GFLOP/s
  - 128KB Act. Buffer → 16x Reduction
  - 128KB Param. Buffer → 32x Reduction
  - Near-data accelerator
Mensa-G: Mensa for Google Edge Models

Based on key characteristics of families, we design three accelerators to efficiently execute inference across our Google NN models.

Google Neural Network Models for Edge Devices: Analyzing and Mitigating Machine Learning Inference Bottlenecks

Amirali Boroumand†, Geraldo F. Oliveira*, Saugata Ghose‡, Xiaoyu Ma§, Berkin Akin§, Ravi Narayanaswami§, Onur Mutlu*†

†Carnegie Mellon Univ.  ‡Stanford Univ.  ‡Univ. of Illinois Urbana-Champaign  §Google  *ETH Zürich

Pascal

Families 1&2 → compute-centric layers
- 32x32 PE Array → 2 TFLOP/s
- 256KB Act. Buffer → 8x Reduction

Jacquard

Families 4&5 → non-LSTM data-centric layers
-16x16 PE Array → 256 GFLOP/s
-128KB Act. Buffer → 16x Reduction
-128KB Param. Buffer → 32x Reduction
- Near-data accelerator

SAFARI

Introduction  TPU and Model Characterization  Mensa Framework  Mensa-G  Evaluation  Conclusion
|   | Introduction                                                                 |
|---|------------------------------------------------------------------------------|
| 2 | Edge TPU and Model Characterization                                          |
| 3 | Mensa Framework                                                              |
| 4 | Mensa-G: Mensa for Google Edge Models                                        |
| 5 | Evaluation                                                                   |
| 6 | Conclusion                                                                   |
Baseline Google Edge TPU accelerator using a high-bandwidth off-chip memory
Energy Analysis

Mensa-G lowers on-chip/off-chip parameter traffic energy by 15.3x by scheduling layers on the accelerator with the most appropriate dataflow and memory bandwidth.

Mensa-G improves energy efficiency by 3.0X compared to the Baseline.
Mensa-G improves **throughput** by **3.1X** compared to the Baseline
More in the Paper

• Details about Mensa Runtime Scheduler

• Details about Pascal, Pavlov, and Jacquard’s dataflows

• Energy comparison with Eyeriss v2

• Mensa-G’s utilization results

• Mensa-G’s inference latency results
Outline

1 Introduction
2 Edge TPU and Model Characterization
3 Mensa Framework
4 Mensa-G: Mensa for Google Edge Models
5 Evaluation
6 Conclusion
Conclusion

**Context:** We extensively analyze a state-of-the-art edge ML accelerator (Google Edge TPU) using 24 Google edge models
- Wide range of models (CNNs, LSTMs, Transducers, RCNNs)

**Problem:** The Edge TPU accelerator suffers from three challenges:
- It operates significantly below its peak throughput
- It operates significantly below its theoretical energy efficiency
- It inefficiently handles memory accesses

**Key Insight:** These shortcomings arise from the monolithic design of the Edge TPU accelerator
- The Edge TPU accelerator design does not account for layer heterogeneity

**Key Mechanism:** A new framework called Mensa
- Mensa consists of heterogeneous accelerators whose dataflow and hardware are specialized for specific families of layers

**Key Results:** We design a version of Mensa for Google edge ML models
- Mensa improves performance and energy by 3.0X and 3.1X
- Mensa reduces cost and improves area efficiency
