ECHO: Compiler-based GPU Memory Footprint Reduction for LSTM RNN Training

Bojian Zheng[1,2], Nandita Vijaykumar[1,3], Gennady Pekhimenko[1,2]

1 Computer Science UNIVERSITY OF TORONTO

2 VECTOR INSTITUTE

3 intel

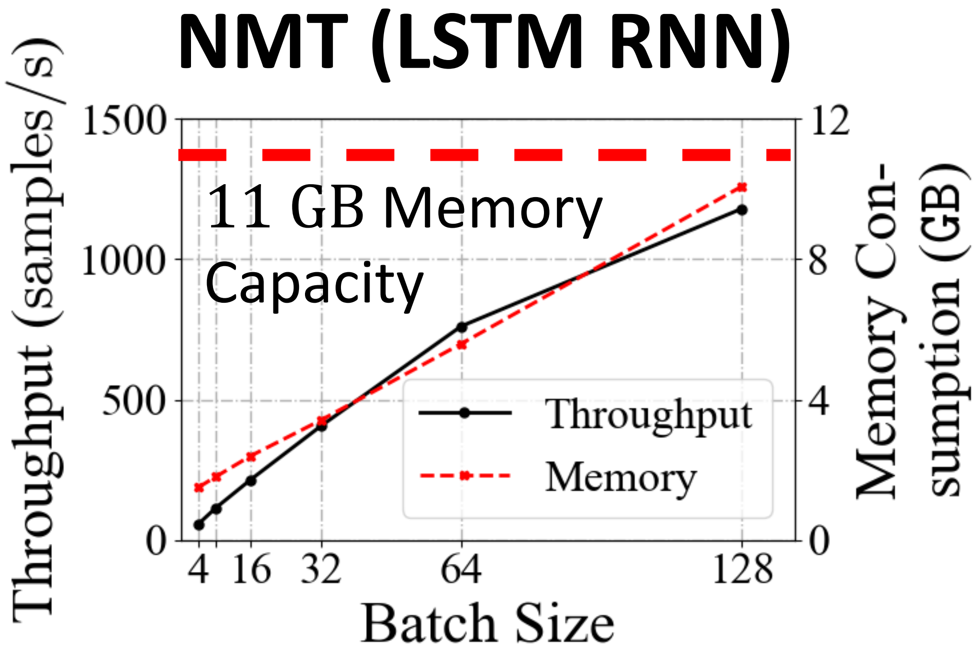
Key Results: $\mathbf{3}\times$ memory footprint reduction → $\mathbf{1.35}\times$ faster training

ECHO and the MXNet GPU memory profiler are both **open-sourced**

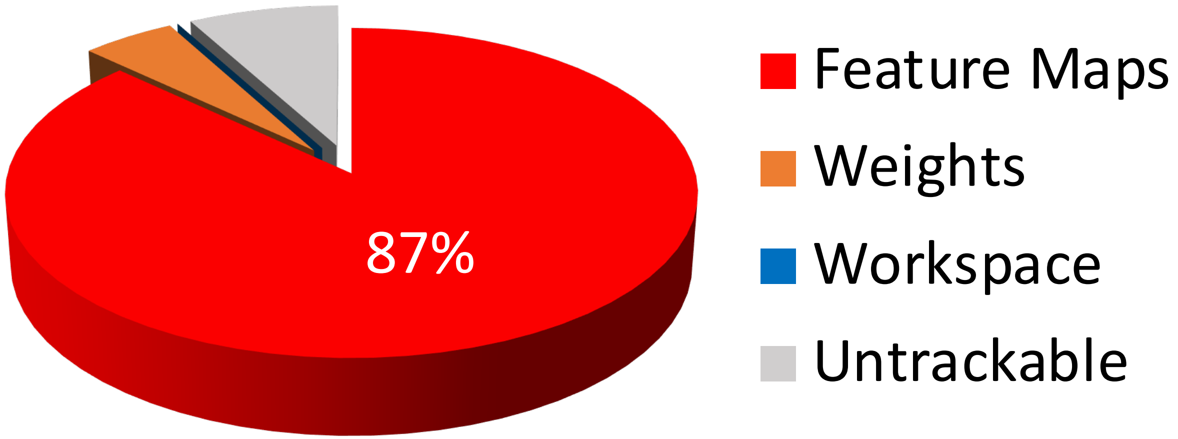
ECHO: https://issues.apache.org/jira/browse/MXNET-1450, GPU Memory Profiler: https://issues.apache.org/jira/browse/MXNET-1404

Why LSTM RNN Training is Inefficient?

Training throughput is limited by the **GPU memory capacity**



Feature maps dominate the GPU memory footprint of the NMT model

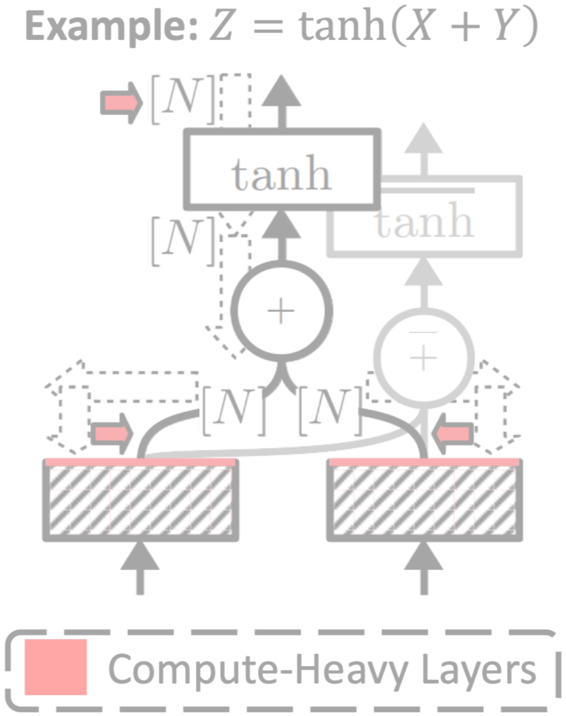


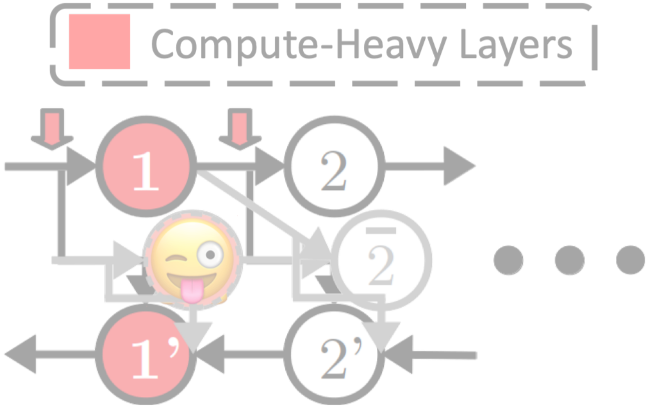
Prior works **fail** to address 2 key challenges:
Estimation of ❶ memory footprint 😕
❷ runtime overhead

Selective Recomputation

ECHO: A Selective Recomputation Graph Compiler Pass

- Open-sourced and integrated in MXNet
  https://issues.apache.org/jira/browse/MXNET-1450
- Fully **Automatic & Transparent**
  - Requires NO changes in the training source code
- Addresses 2 key challenges: Estimation of
  1. memory footprint: Bidirectional Dataflow Analysis
  2. runtime overhead: Layer-Specific Optimizations

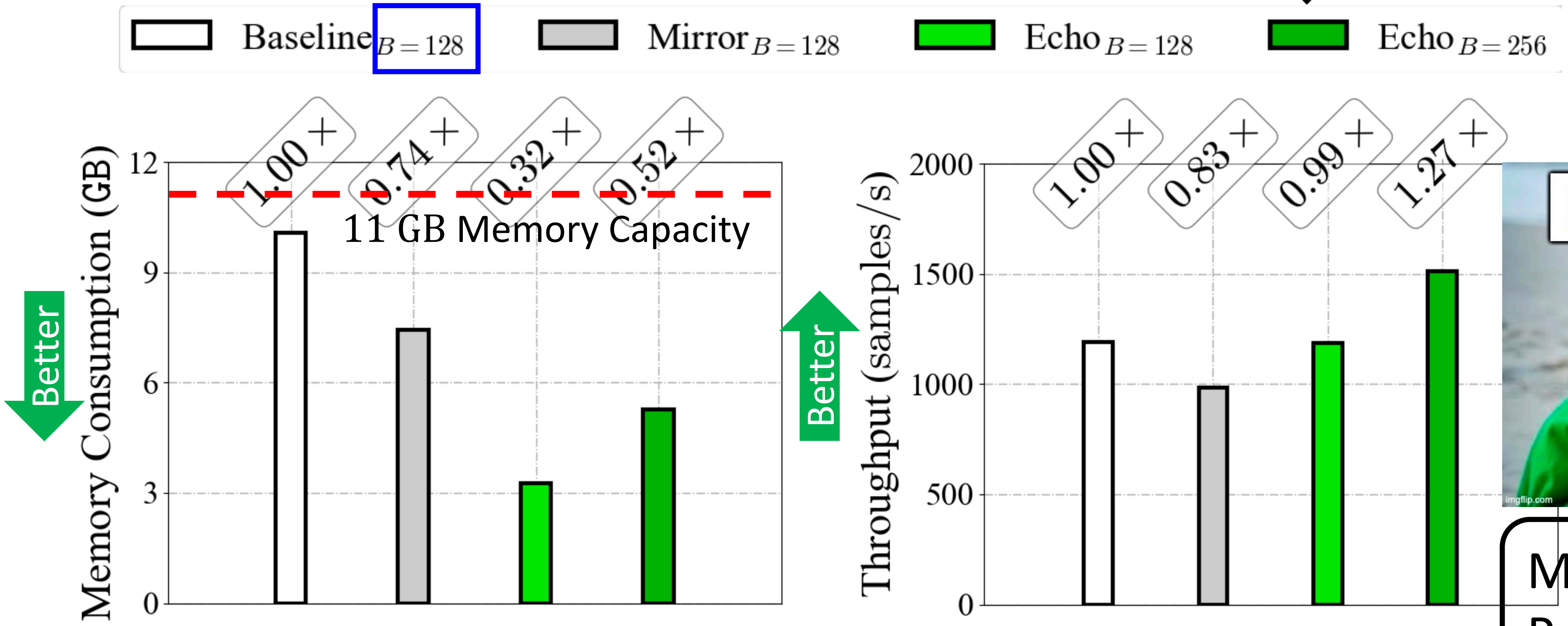


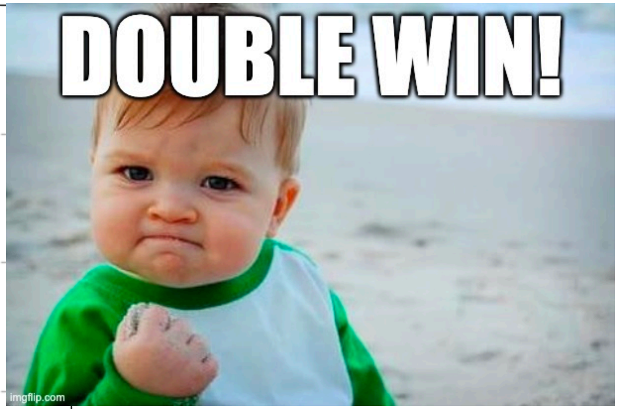


ECHO ’s Effect on Memory and Performance

Baseline: NO Recomputation, **Mirror**: T. Chen et al.[1], **ECHO**: Our Work

IWSLT15 EN-VI Dataset, Single RTX 2080 Ti GPU 2×





Memory Footprint ↓
Performance ↑