上海交通大學
SHANGHAI JIAO TONG UNIVERSITY

SME: ReRAM-based Sparse-Multiplication-Engine to Squeeze-Out Bit Sparsity of Neural Network

Fangxin Liu (Speaker)

Wenbo Zhao, Zhezhi He, Zongwu Wang, Yilong Zhao, Yang Tao, Xiaoyao Liang, Naifeng Jing and Li Jiang*

Shanghai Jiao Tong University

2022年2月28日

ICCD-39
Oct. 26, 2021

上海交通大學
SHANGHAI JIAO TONG UNIVERSITY

上海交通大學
SHANGHAI JIAO TONG UNIVERSITY

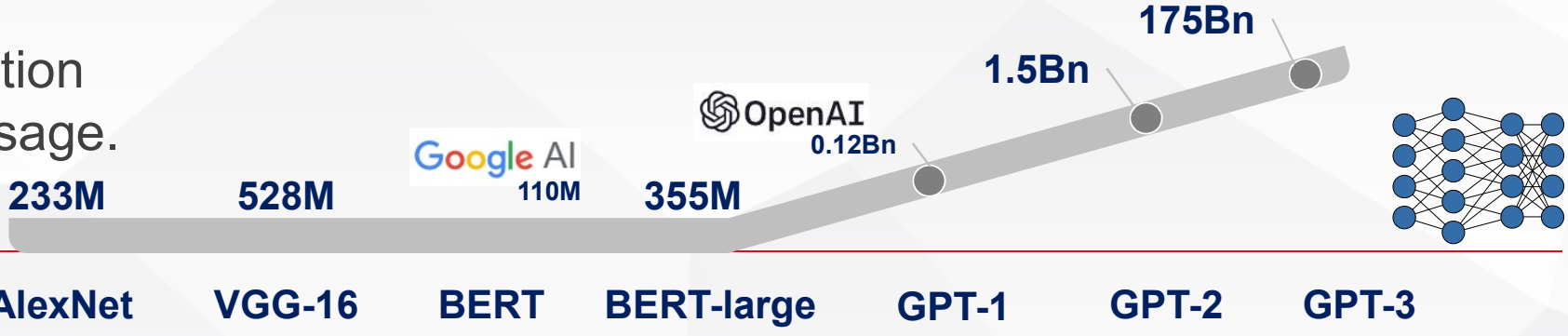
目录

01

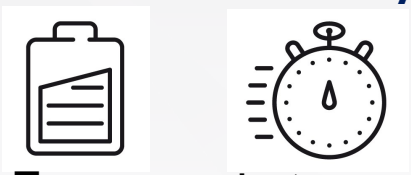
Background & Motivation

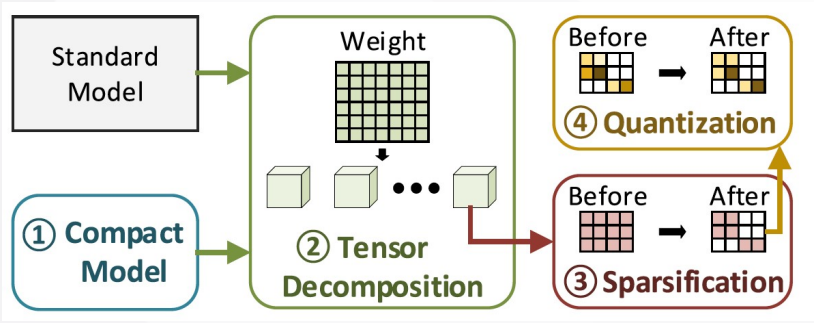
DNN Model Compression

Enormous computation cost and memory usage.



Challenges

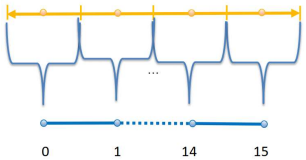




Weight Quantization & Network Sparsification

Fewer parameters and less computation cost

Floating Point Values



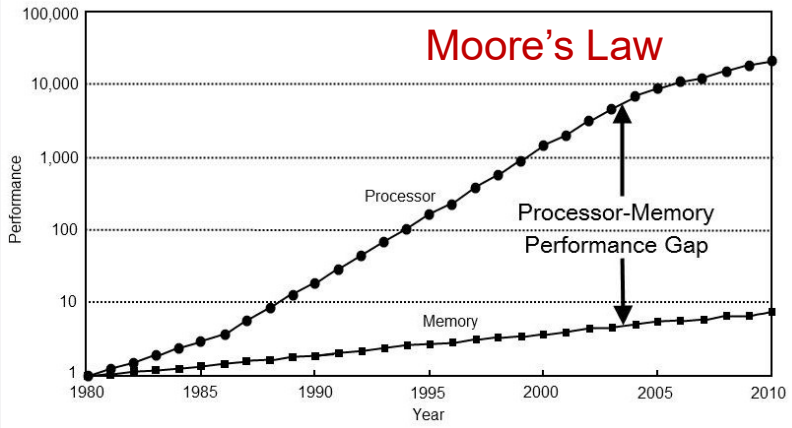
4-bit Quantization

Column/Row/Block-wise Sparsity

DNN Accelerator Limitation

Memory Wall

Memory Access Speed Limitation



Memory is the key to enable true intelligence

Memory Access Energy Consumption

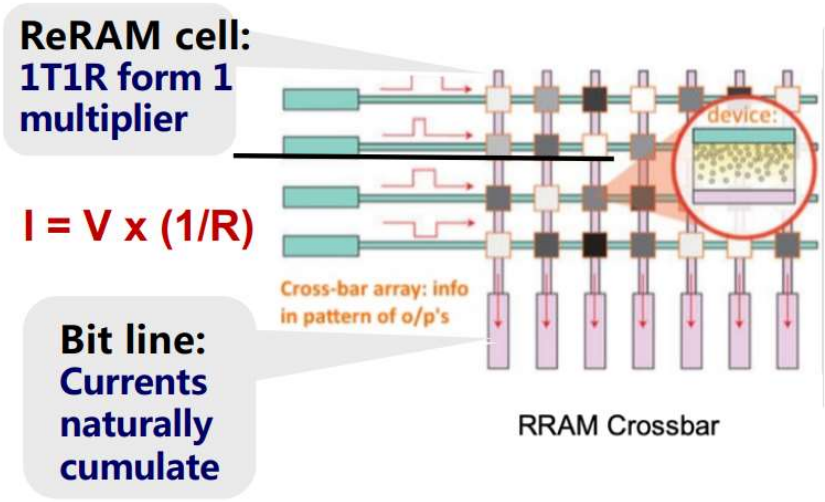
- **DDR4 DIMMs:** 320 pJ/Byte
- **In-package HBM DRAM**: 64pJ/Byte
- **In-processor SRAM**:

  6pJ/bit for 8Mbit → 47pJ/bit for 64Mbit
- **In-processor Crossbar ReRAM**: <0.5pJ/bit

In Memory computing

Background: ReRAM-Based DNN Accelerator

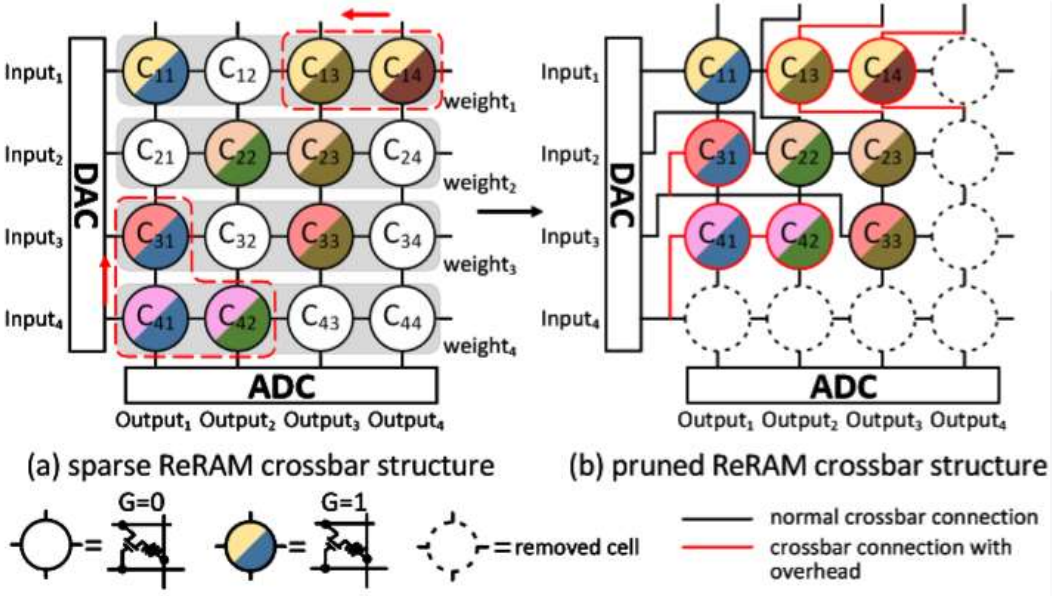
- Saves the weights on non-volatile resistive random-access memory (ReRAM).
- Operates Multiply-and-Accumulate (MAC) operations by gathering the analog currents in verticle bit-lines.
- Uses Digital-Analog Converters (DACs) and Analog-Digital Converters (ADCs) to communicate between digital peripheral circuit and analog ReRAM crossbar.



Weakness & Motivation

Structural-coupling problem:

manifests itself as the inability to freely skip the multiplication of zero operands because weight-bits in the same crossbar-row share the same input, and the current derived by multiplication in cells are accumulated in the same crossbar-column.



Structural pruning methods avoid this problem by pruning the weights in a granularity that the whole crossbar-column (or -row) can be removed at the cost of extra peripheral circuits

Coarse pruning granularity will inevitably modify weight values and thus requires finetuning to retrieve accuracy.

02

Algorithm Design

SME

Quantization & Encoding

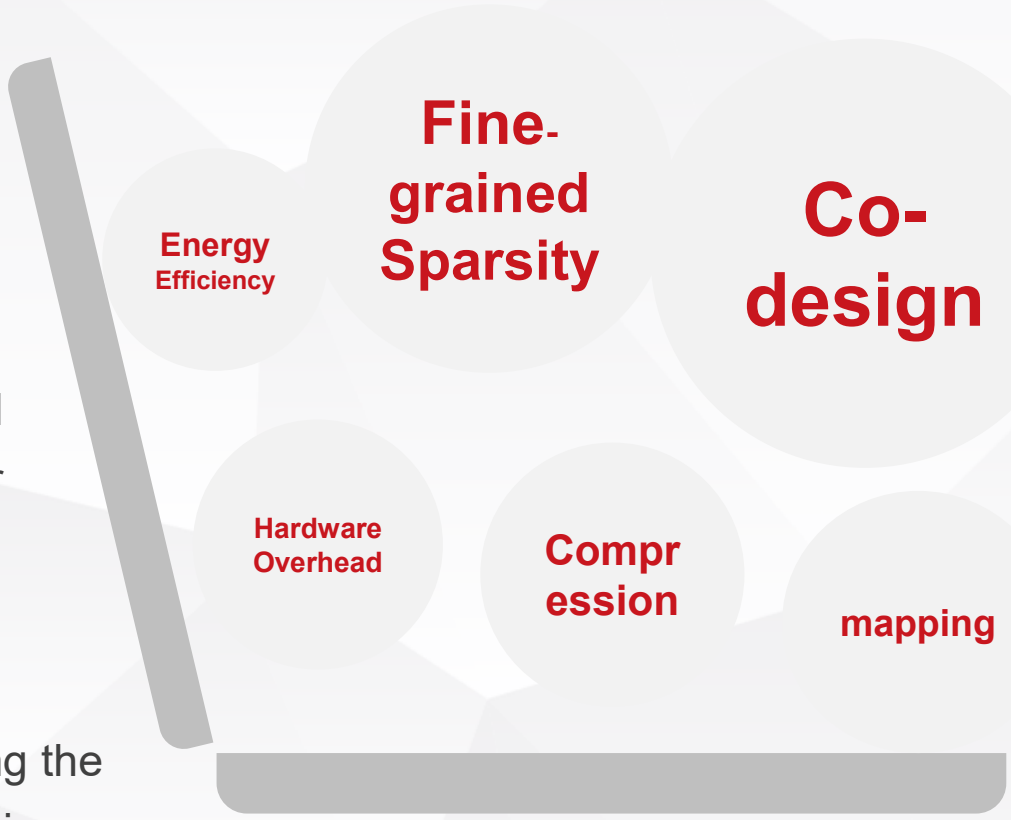
increase and accumulate the bit-level sparsity in a codeword

Bit-slicing

decouple the crossbar structure and aggregate a large amount of regular sparsity

Bit-wise Squeeze-out

The essence is row swapping among the crossbar group, but without introducing either overhead or large accuracy loss.



Challenges of Our Proposal

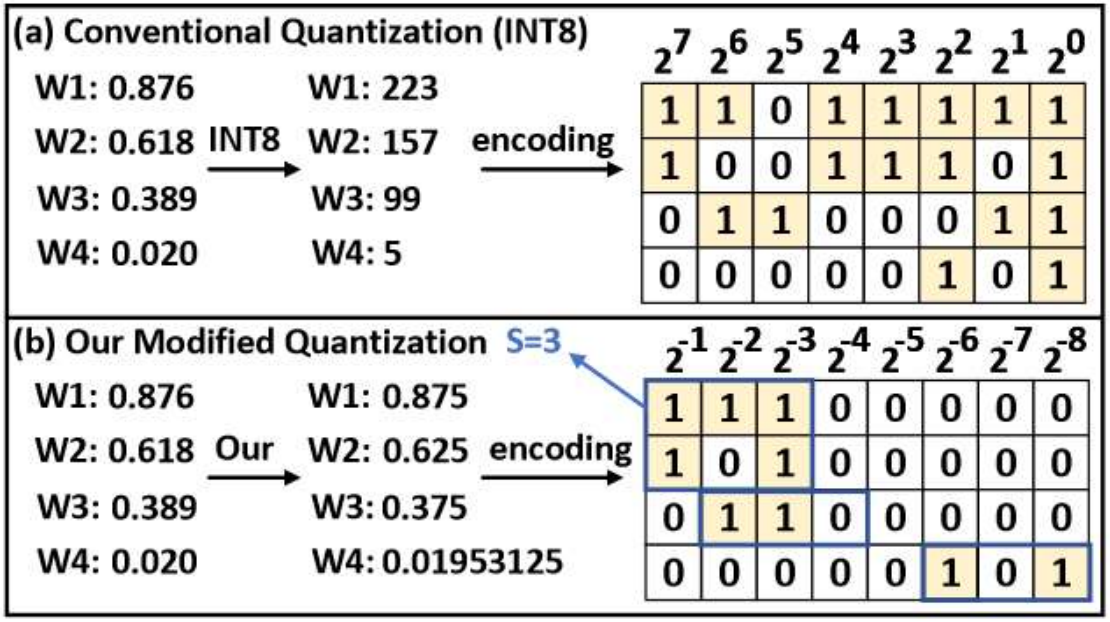
How to exploit the fine-grained sparsity with light overhead?

- The fundamental limit of exploiting the sparsity is because **the data mapping and the VMM computation are tightly coupled with the crossbar structure**

How to support efficient SME algorithm?

- We design the architecture to efficiently support our algorithm through well-designed crossbars with the peripheral circuit.

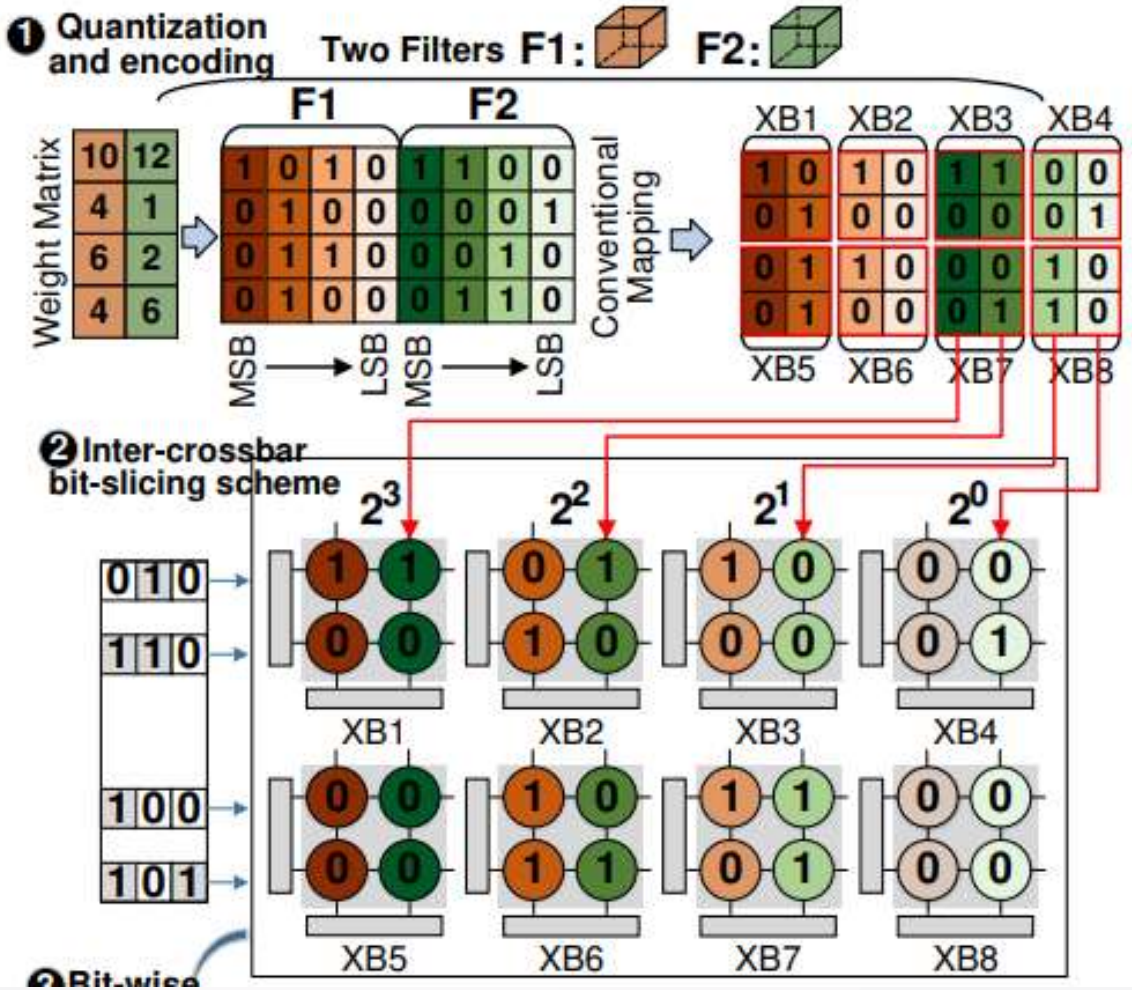
Overview of SME algorithm



Quantization and Encoding Scheme

We quantize the weights into sum of power-of-twos, whose **exponent are among S consecutive integers. This results in a regular sparse pattern.**

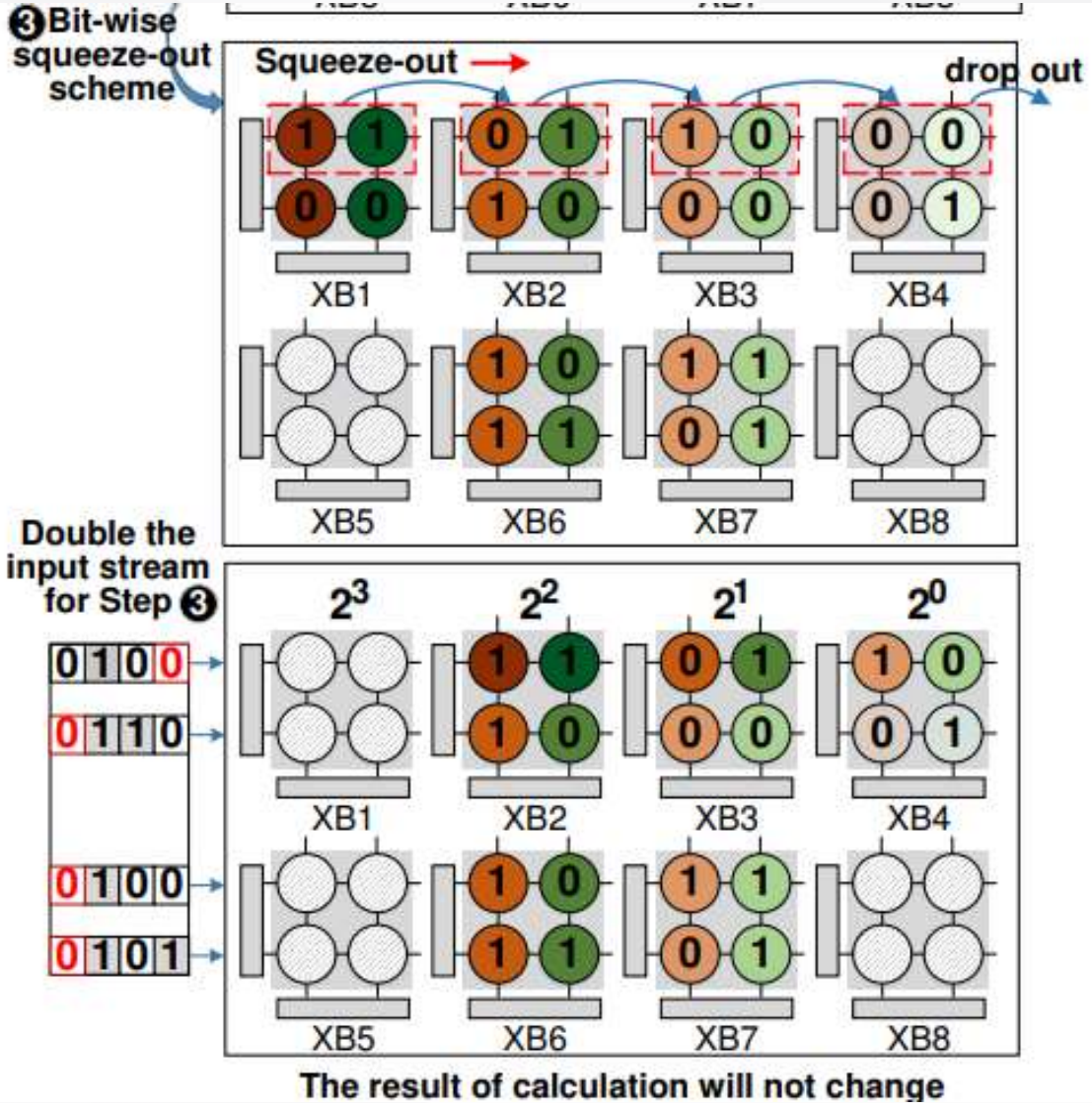
Overview of SME algorithm



Inter-crossbar Bit-slicing Scheme

To decouple the crossbar structure, we propose the inter-crossbar bit-slicing scheme. The key idea of bit-slicing is to **map the same bit of quantized weights into the same bit crossbar.**

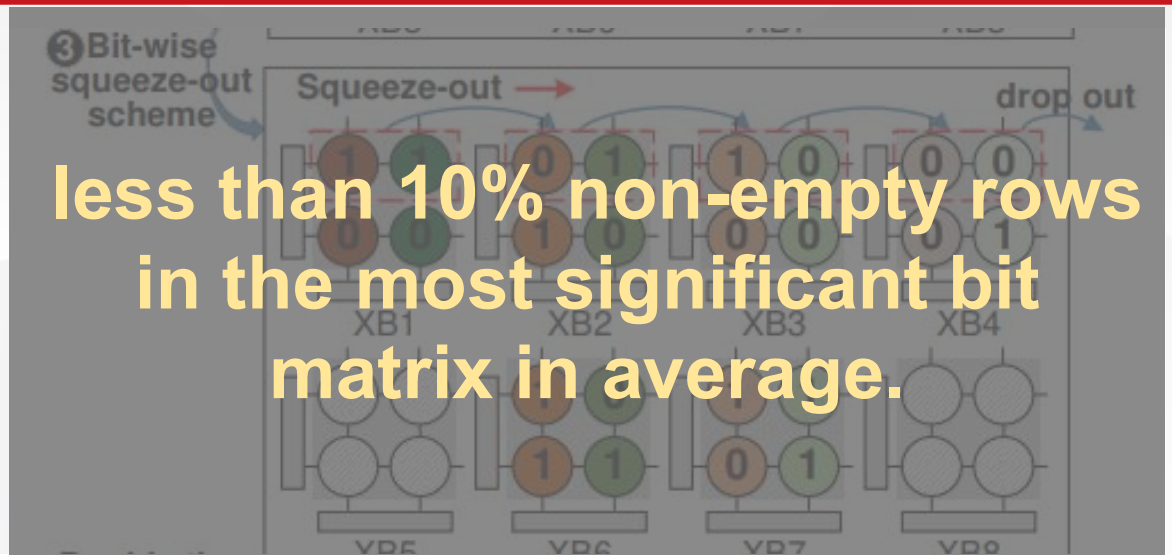
Overview of SME algorithm



Bit-wise Squeeze-out Scheme

We squeeze the crossbar-rows containing non-zeros in preceding XBs to the subsequent XBs until these rows in tailing XBs are dropped out.

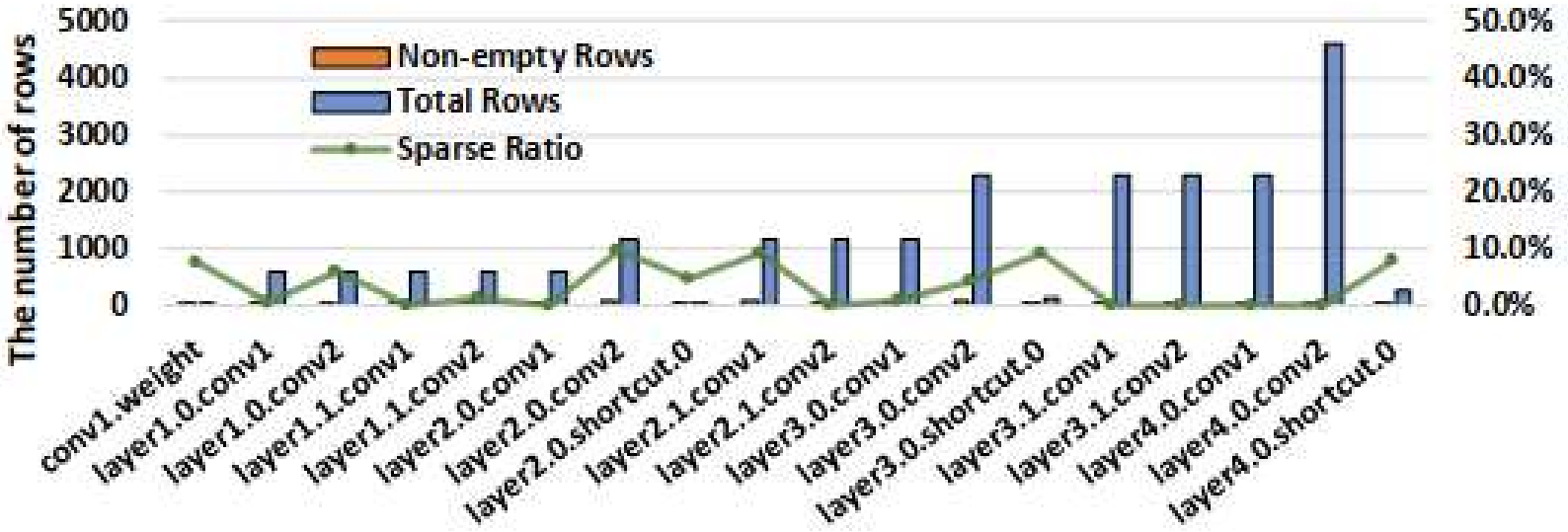
Overview of SME algorithm



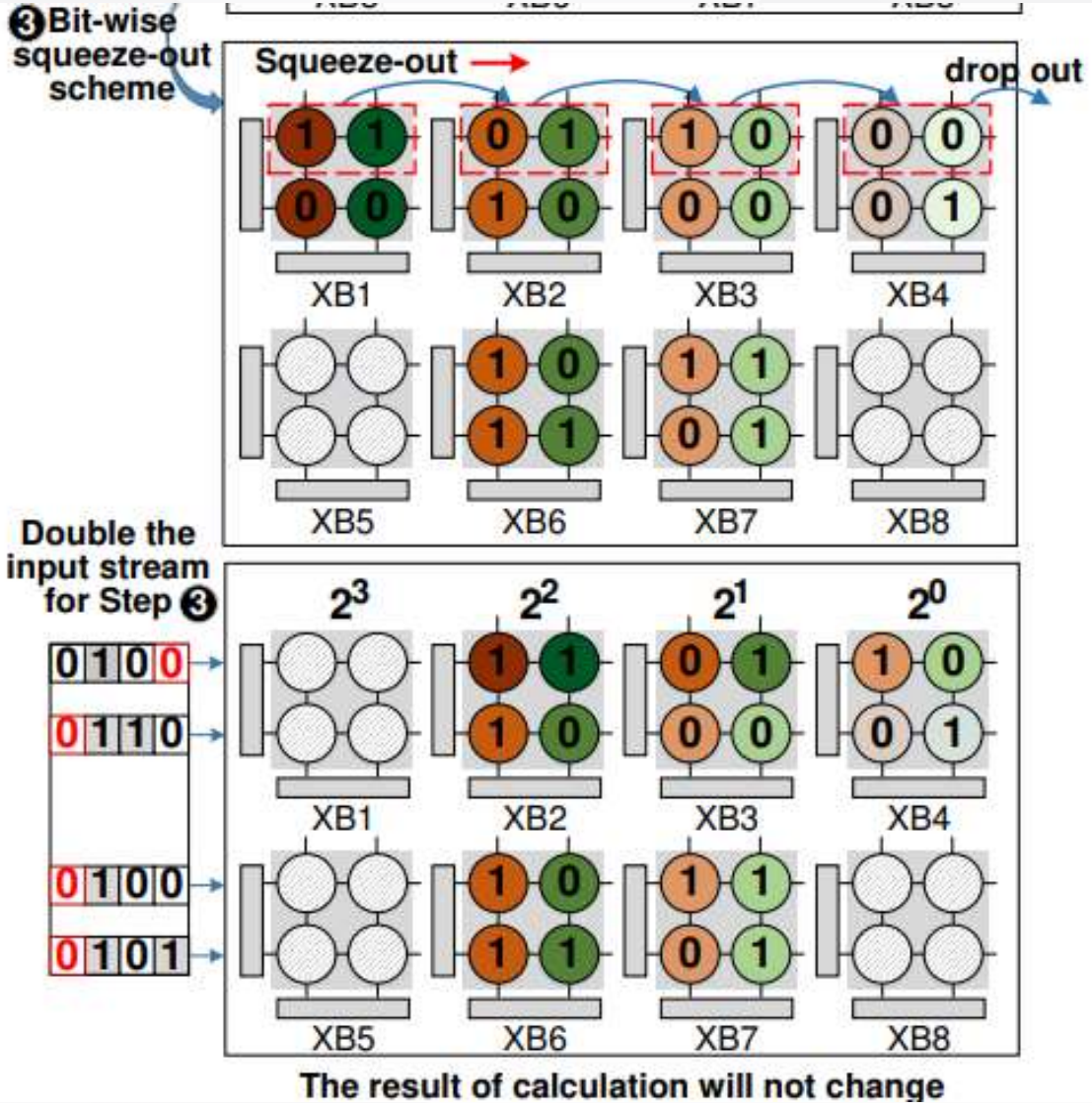
less than 10% non-empty rows in the most significant bit matrix in average.

Bit-wise Squeeze-out Scheme

We squeeze the crossbar-rows containing non-zeros in preceding XBs to the 3s until these XBs are dropped



Overview of SME algorithm



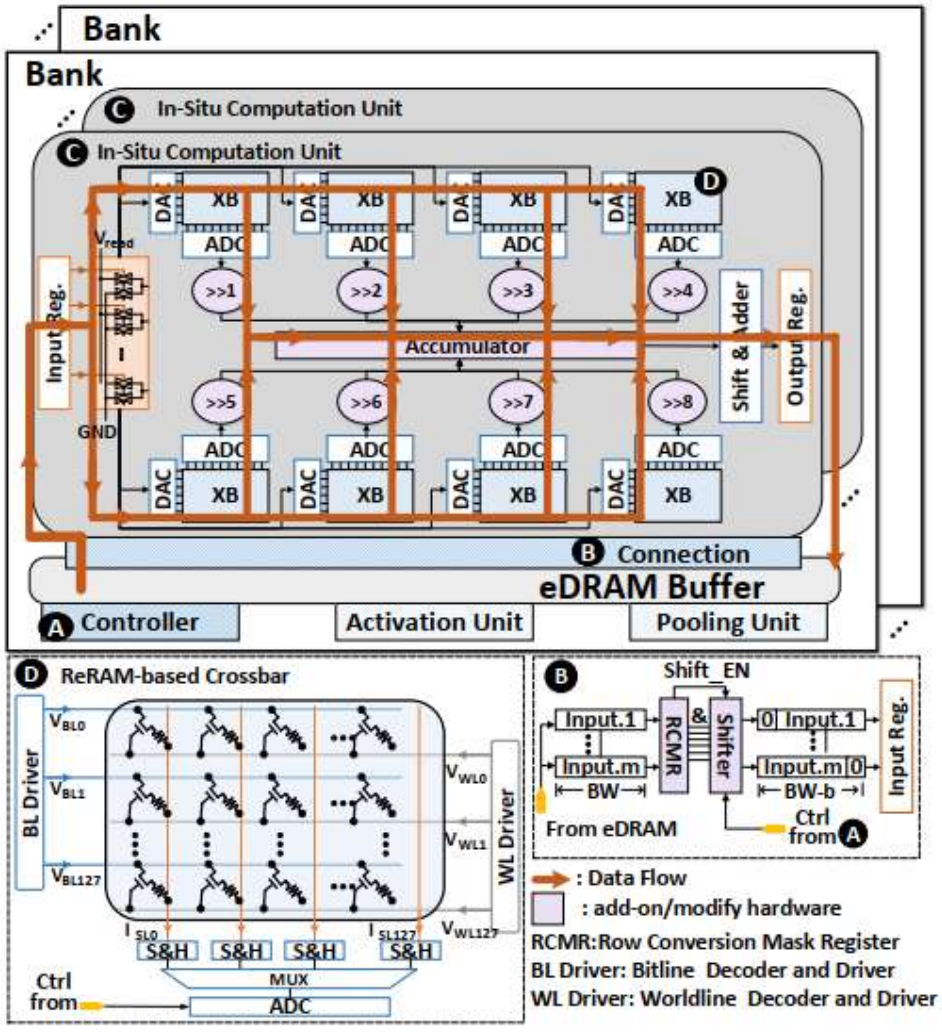
Bit-wise Squeeze-out Scheme

We squeeze the crossbar-rows containing non-zeros in preceding XBs to the subsequent XBs until these rows in tailing XBs are dropped out.

03

Architecture Design

Overview of Architecture



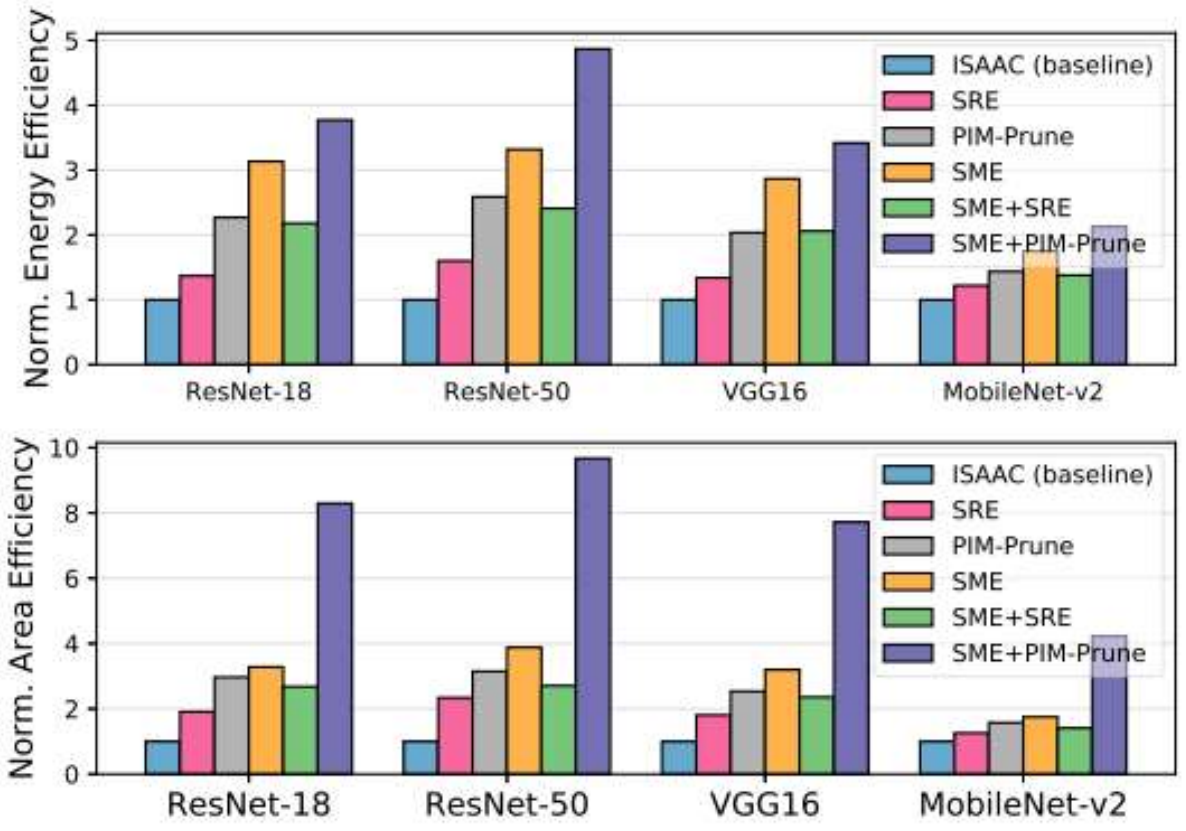
SME Architecture:

aiming at inference in edge devices. The SME add-on hardware implements, including simple modifications to the existing crossbar peripheral circuits, which is easier to manufacture than integrate complex logic into the chip.

04

Evaluation

Evaluation



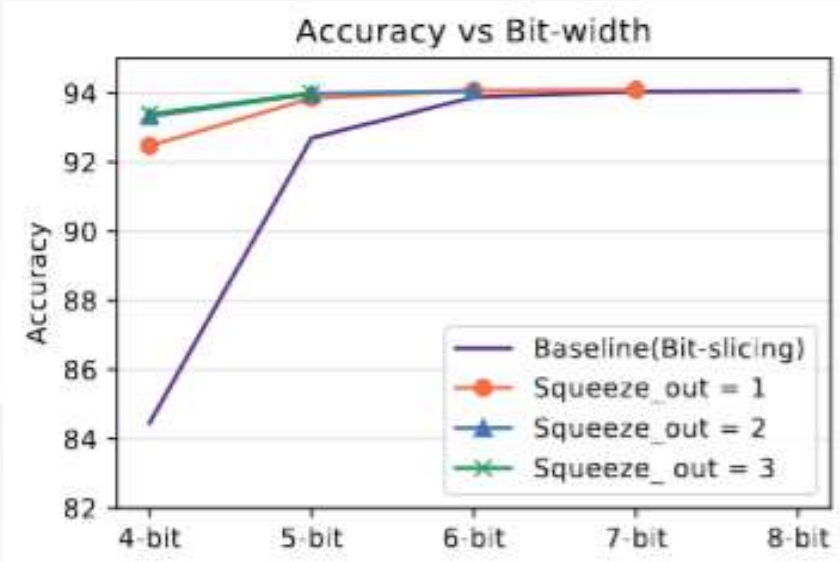
Energy- and Area-Efficiency:

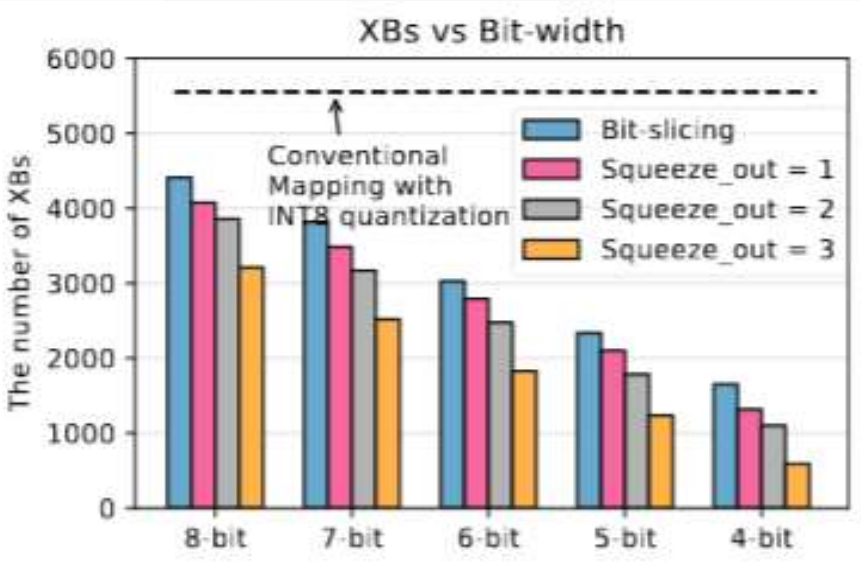
- Up to 2.3 X energy efficiency
- Up to 6.1 X area efficiency

Evaluation

Varied squeeze-out schemes with crossbar resource:

• We use the squeeze-out scheme to reduce the number of cells representing weights far better than directly reducing because the MSBs are more critical than the LSBs.



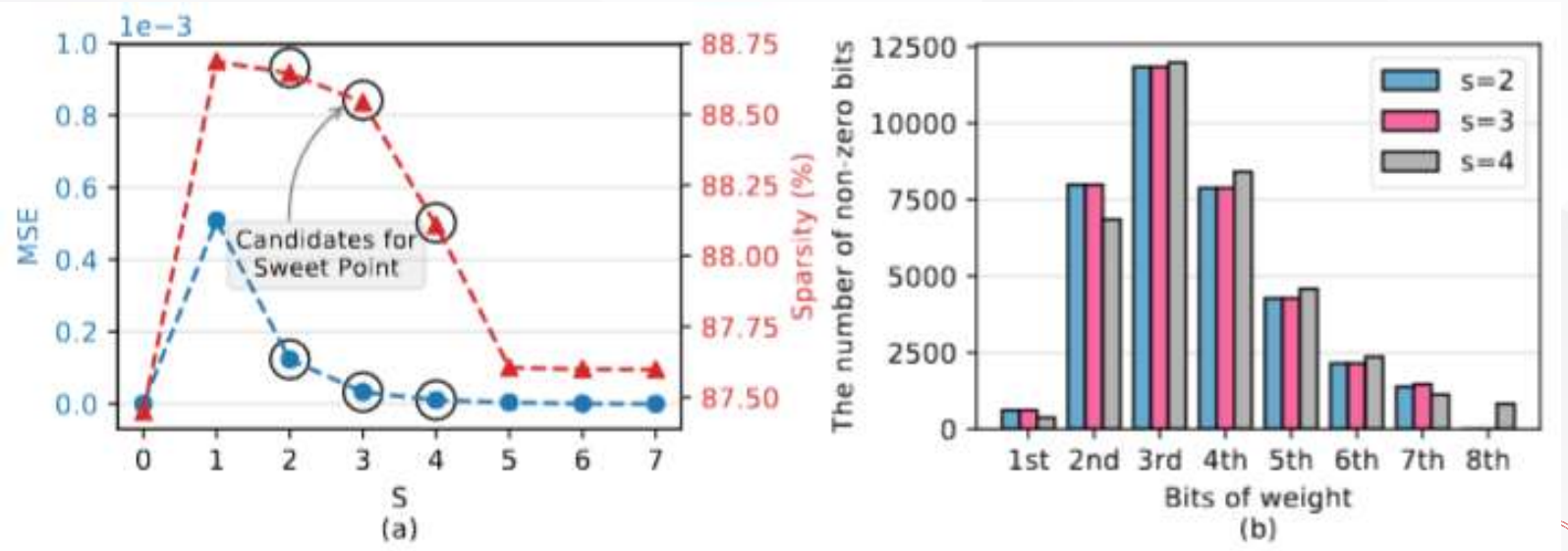


Evaluation

Sweet-spot for the size of consecutive region containing '1':

• We combine with the overall sparsity and the bit-level sparse distribution.

We can find that $S=3$, SME achieves an optimal point for ResNet-18.



05

Conclusion

Conclusion

A novel SME algorithm

- decouples the hardware dependence of multiplication
- release the sparse cells in the crossbars for higher energy-/area-efficient inference

An efficient SME architecture

- Well-designed crossbars with the peripheral circuits
- Efficiently support the fine-grained sparsity generated by the our algorithm

Keep high accuracy while gaining large improvement in terms of energy and area

Thanks for Listening

WeChat

lfx920701

饮 水 思 源　爱 国 荣 校