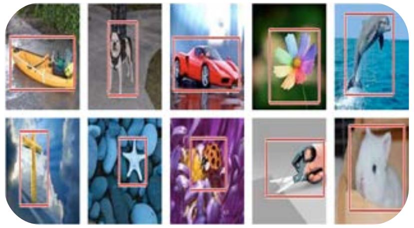
Exploring Energy and Accuracy Tradeoff in Structure Simplification of Trained Deep Neural Networks*

Boyu Zhang, Azadeh Davoodi, Yu Hen Hu
Department of Electrical & Computer Engineering
University of Wisconsin Madison

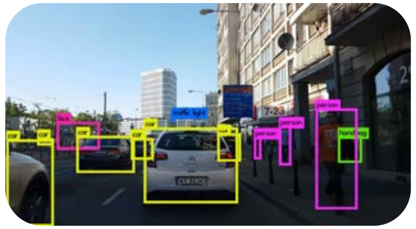
Outline

- Challenges in realization of modern Deep Neural Networks (DNNs)
- Recent advances on efficient realization of DNNs
- Our contributions and approach
- Results and conclusions

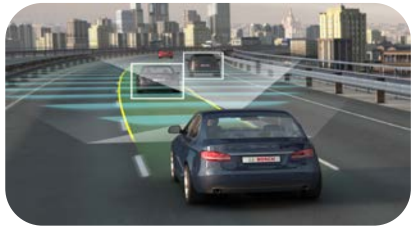
Challenges in Modern Deep Neural Networks



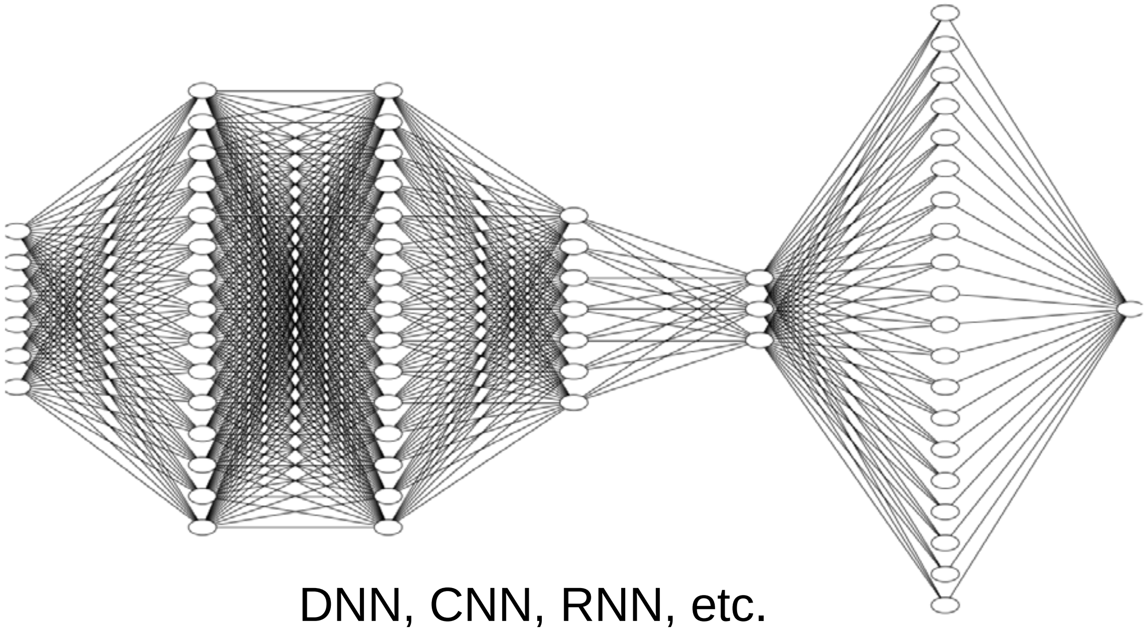
Object Localization



Object Detection



Autonomous Vehicle



DNN, CNN, RNN, etc.

Natural Language Processing

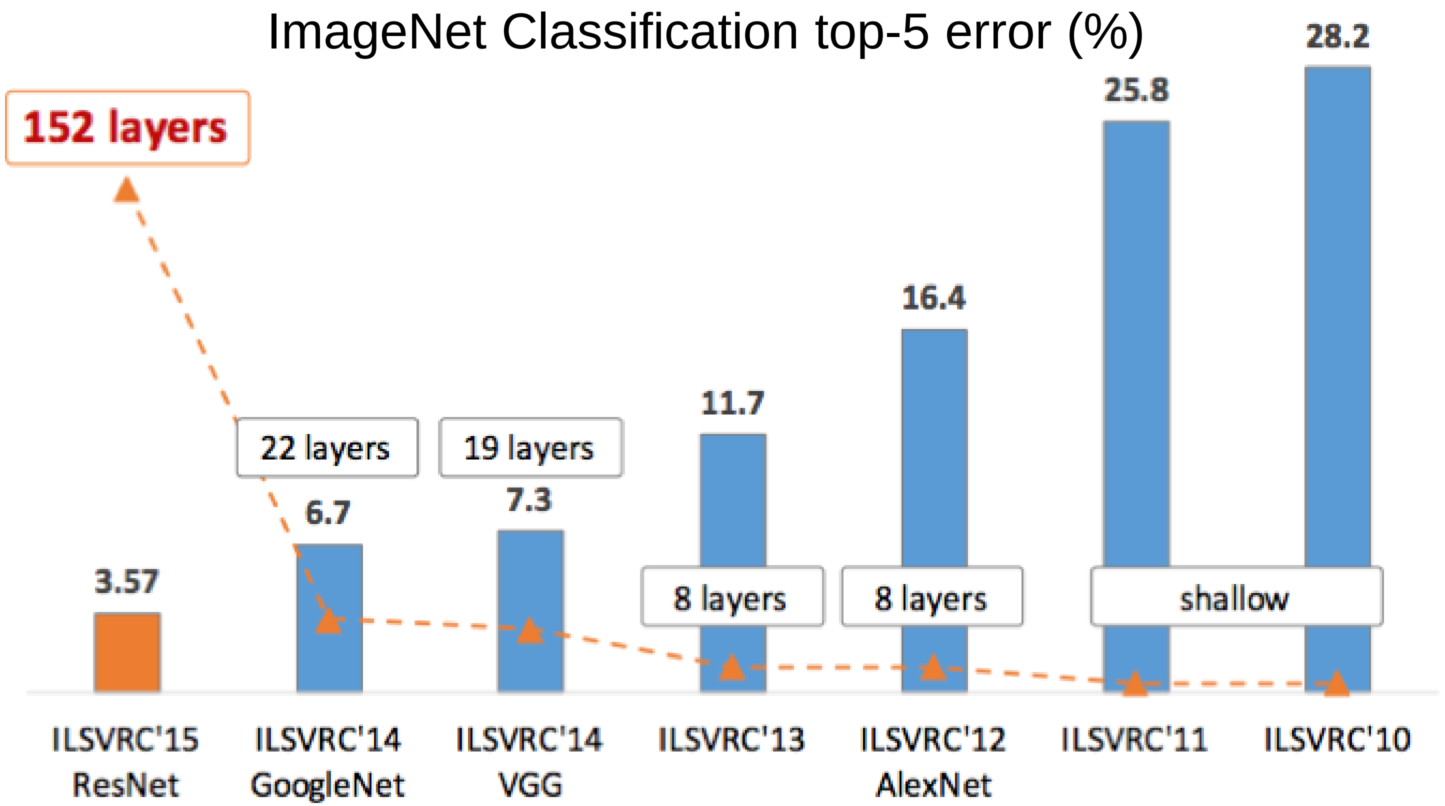
Machine Translation

Game AI

Image Caption Generation

Challenges in Modern Deep Neural Networks

Revolution of Depth



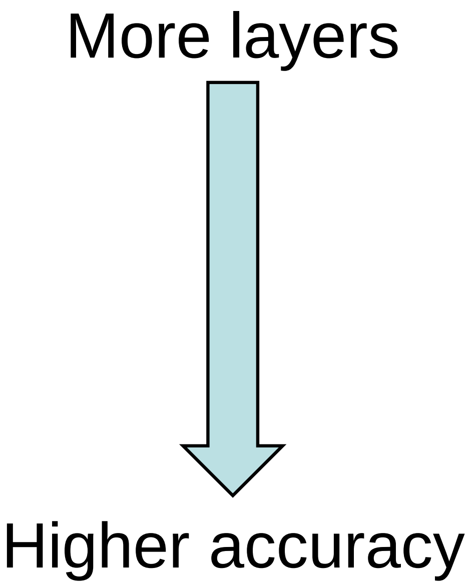
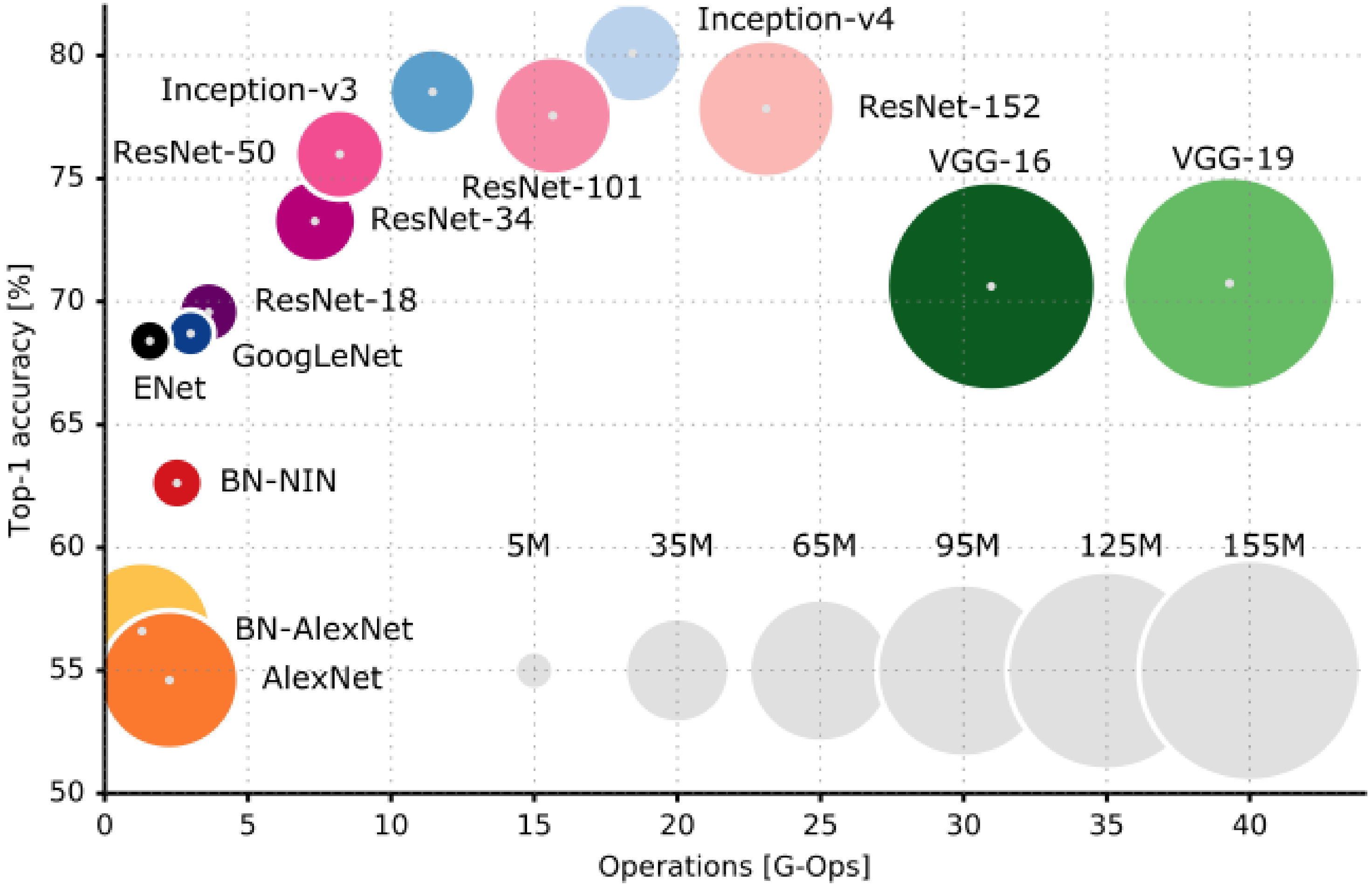


Image Credit: He, Kaiming, et al. "Deep residual learning for image recognition." Proceedings of the IEEE conference on computer vision and pattern recognition. 2016.

Challenges in Modern Deep Neural Networks



In general, DNNs are expensive, in terms of memory, computation, and energy.

Challenge when implementing on embedded systems and mobile devices.

Need to make it affordable.

Image Credit: Canziani, Alfredo, Adam Paszke, and Eugenio Culurciello. "An analysis of deep neural network models for practical applications." arXiv preprint arXiv:1605.07678 (2016).

Outline

- Challenges in realization of modern Deep Neural Networks (DNNs)
- Recent advances on efficient realization of DNNs
- Our contributions and approach
- Results and conclusions

Recent Advances in Design of Efficient DNNs

- Reducing the computation cost
  - Quantization of weights and activation, stochastic computing, …

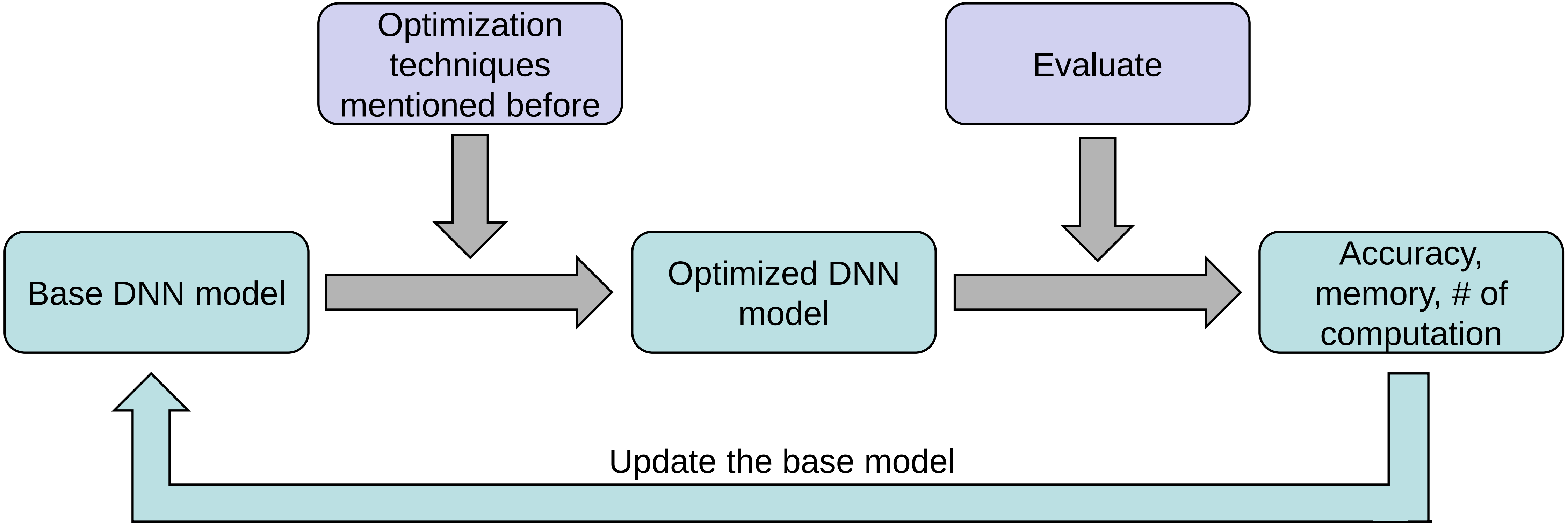
  (The structure of the network is intact and may not be optimal)
- Network structure simplification
  - Low-rank approximation of network’s layers
  - Reducing number of weights: edge pruning, brain damage, weight sharing, …

  (Results in sparse weight matrix, requires special hardware to fully utilize its potential)
- New techniques for designing the network structure
  - Example: SqueezeNet, ResNet, Inception module, …

  (Almost Art! Mostly manual process)

Recent Advances in Design of Efficient DNNs

Conventional design flow: **ENERGY is missing!**



Outline

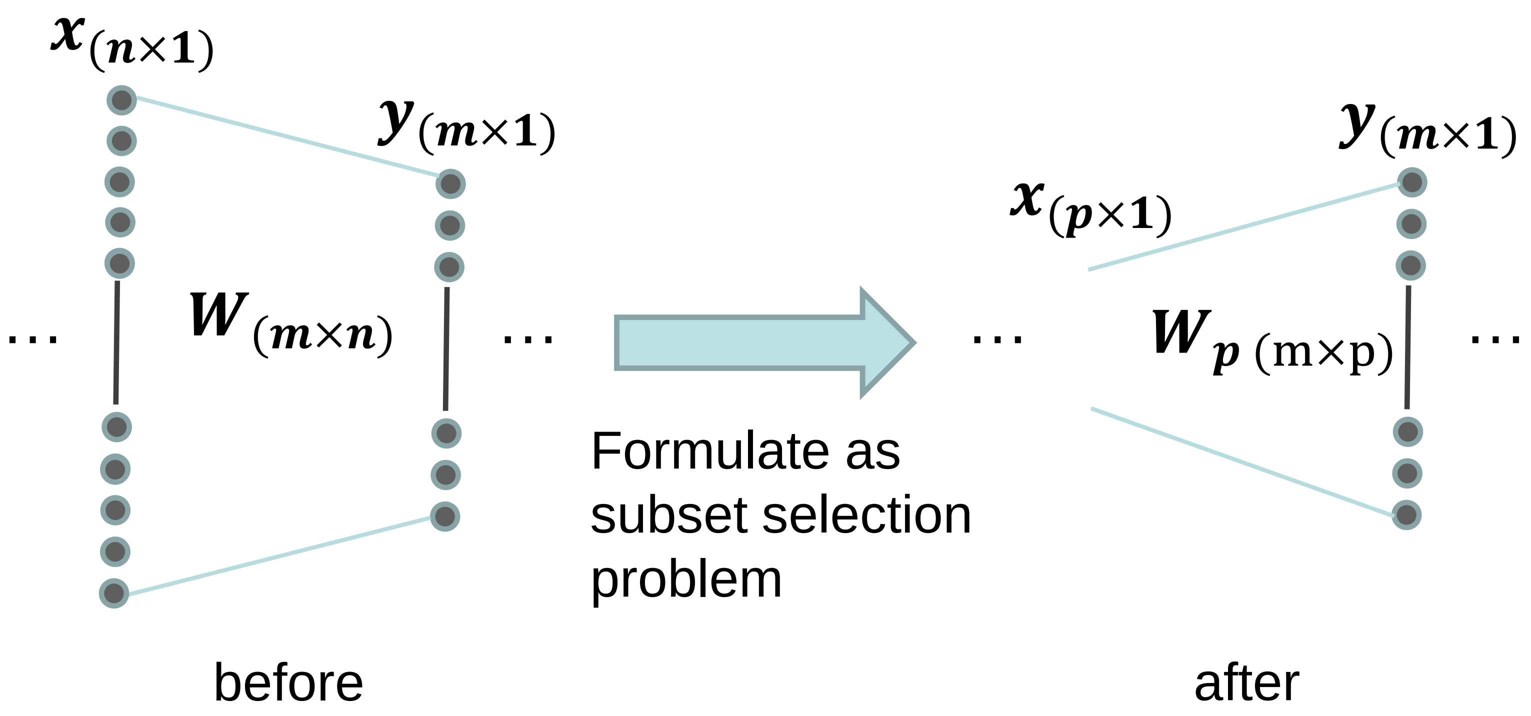
- Challenges in realization of modern Deep Neural Networks (DNNs)
- Recent advances on efficient realization of DNNs
- Our contributions and approach
- Results and conclusions

Our contributions

- New technique for DNN structure simplification based on eliminating redundant neurons
- Our technique won’t require retraining the DNN and is compatible with other prior work on DNN simplification
- Explicitly minimize energy during DNN structure simplification
- Show that considering energy-accuracy tradeoff impacts how the network is simplified

Overview of Our Work

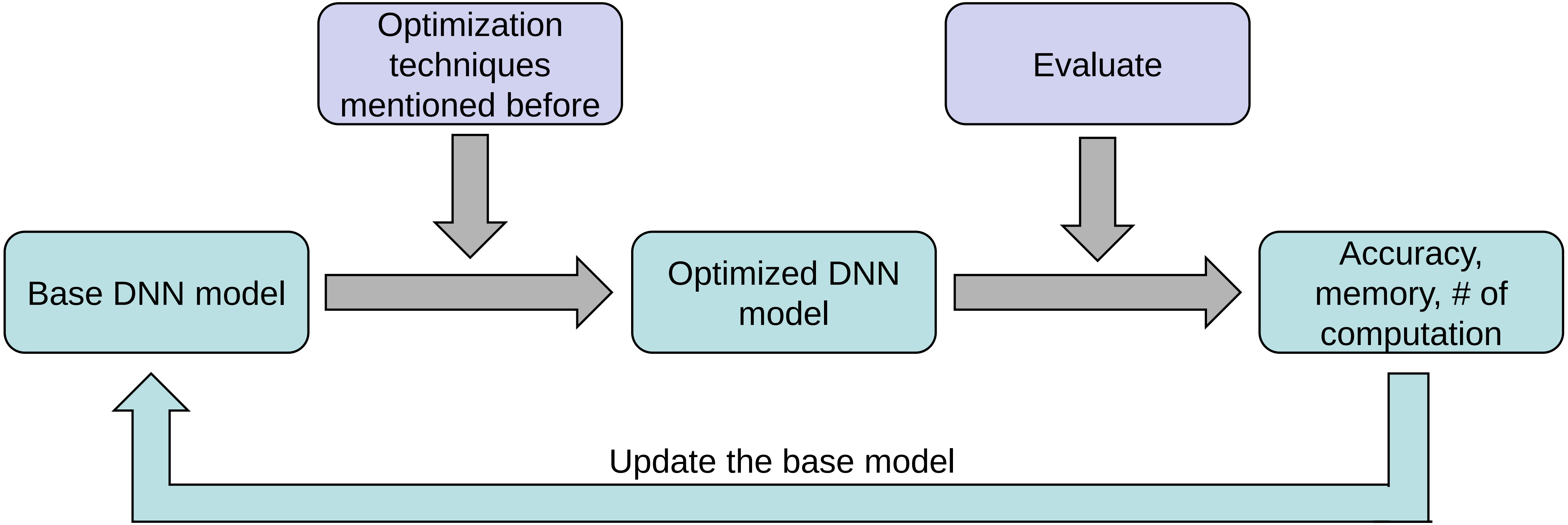
Elimination of neurons in a considered layer:



Our technique computes the new updated weights

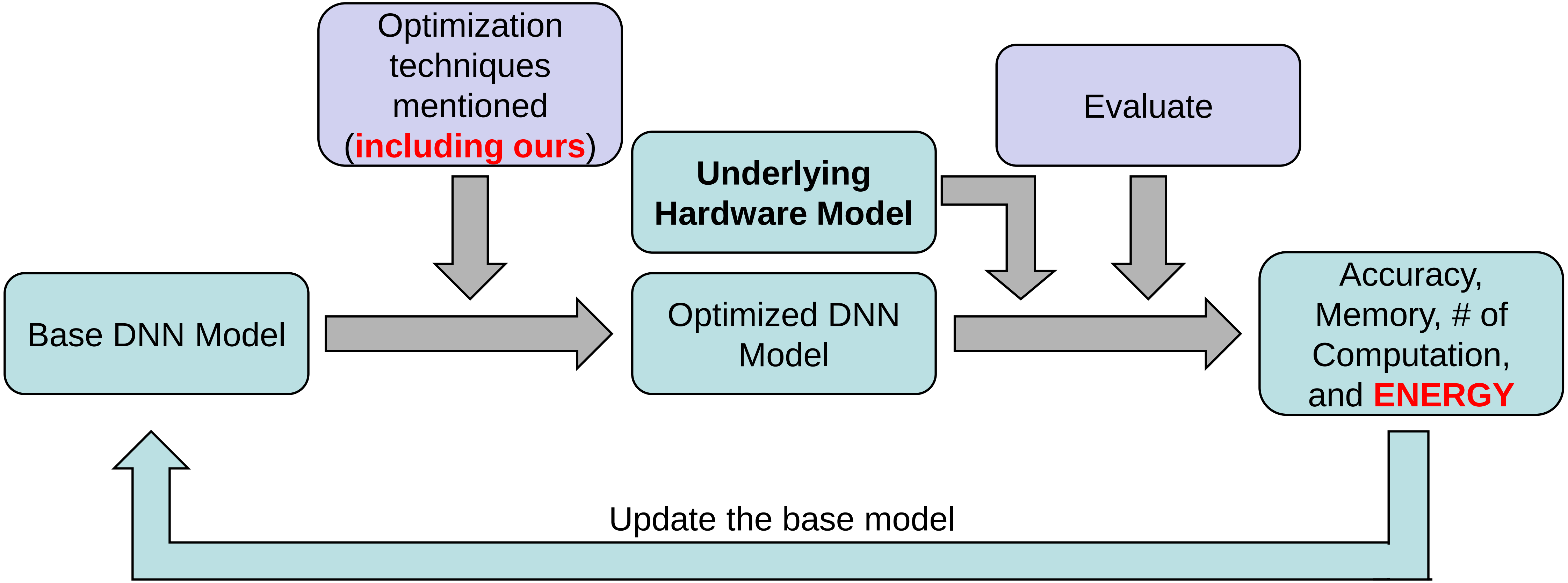
Overview of Our Work

Conventional design flow: **ENERGY is missing!**

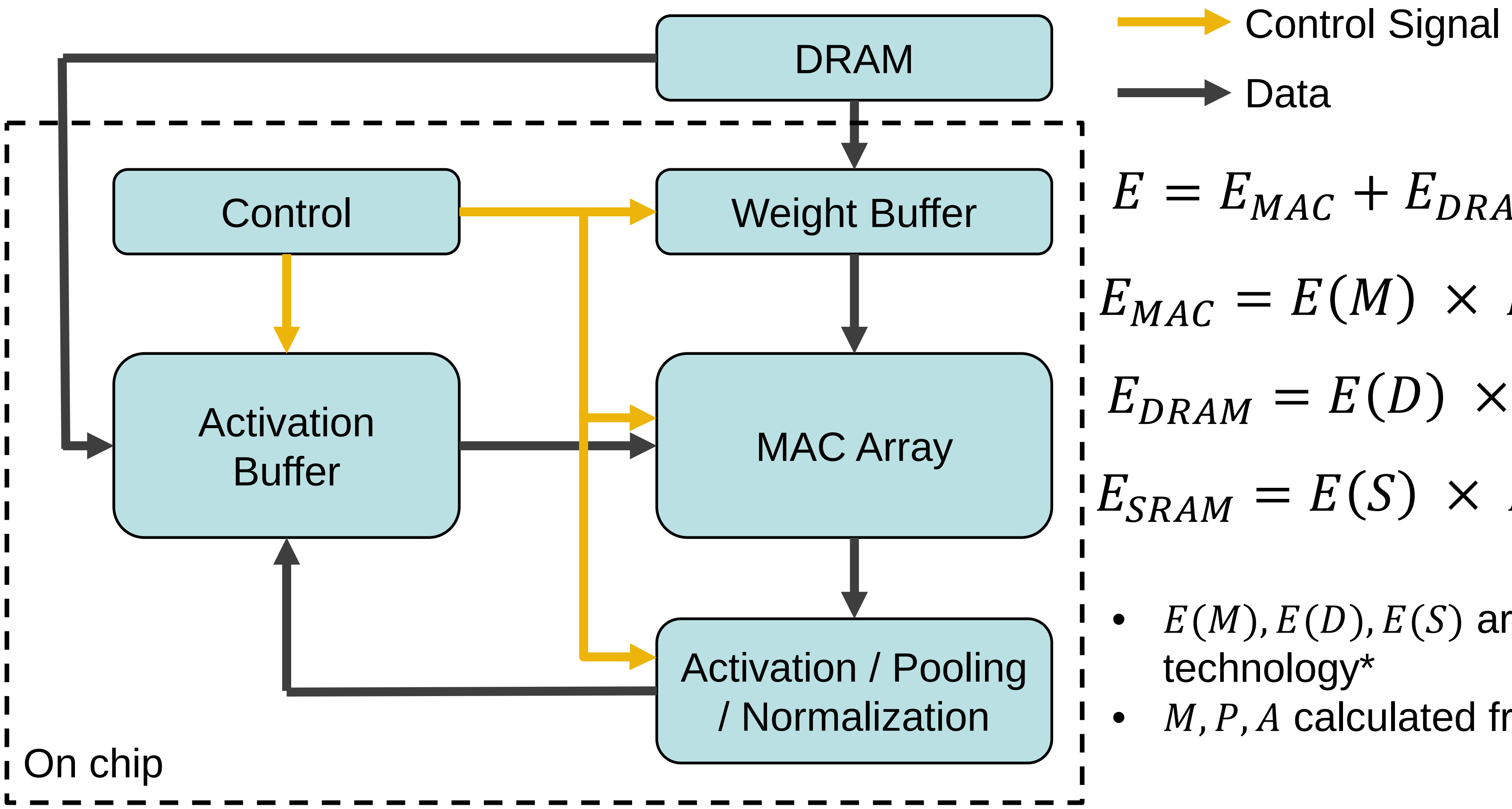


Overview of Our Work

Our design flow: **ENERGY is incorporated!**



Our Energy Model



$$E = E_{MAC} + E_{DRAM} + E_{SRAM} + E_{REST}$$

$$E_{MAC} = E(M) \times M$$

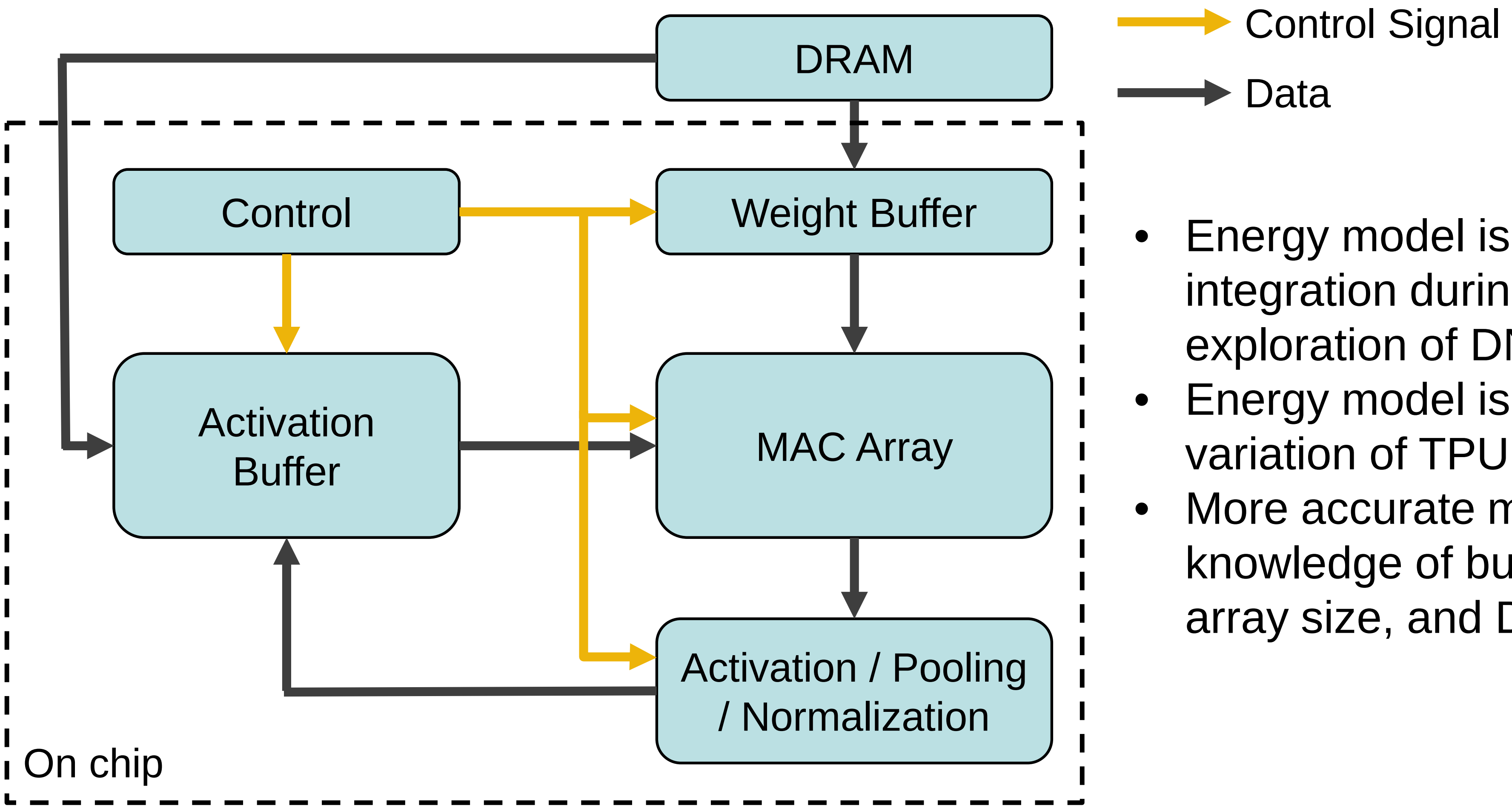
$$E_{DRAM} = E(D) \times (P + Image\ Size)$$

$$E_{SRAM} = E(S) \times P + 2E(S) \times A$$

- $E(M), E(D), E(S)$ are known for given technology*
- M, P, A calculated from the DNN model

* Mark Horowitz. Energy table for 45nm process, Stanford VLSI Wiki.
* Han, Song, et al. "Learning both weights and connections for efficient neural network." *Advances in Neural Information Processing Systems*. 2015.

Our Energy Model



- Energy model is convenient for integration during design space exploration of DNN
- Energy model is simplified variation of TPU [ISCA17]
- More accurate model requires knowledge of buffer sizes, MAC array size, and DNN model size

Neuron Elimination Algorithm

Algorithm 1

procedure ReduceDimension($\mathbf{X}_{n\times K}$, layer ℓ)
  p = rank($\mathbf{X}$); $E_{cur} = 10^{-6}$
  Do {
    Select p rows of $\mathbf{X}$ using Algo. 2
    Compute $\mathbf{W}_p$ using Eq. 6
    Generate simplified network N as in Fig. 3
    Record accuracy degradation ϵ and energy E_{cur} of N
    $p = p - 1$
  }
  While($p \neq 0$ and $\epsilon \leq$ degradation threshold)
  Generate accuracy vs energy tradeoff of stored configurations
end procedure

Algorithm 2

procedure FindRepresentativeRows($\mathbf{X}_{n\times K}$, p)
  Perform SVD decomposition on $\mathbf{X} = \mathbf{U\Sigma V}^T$
  Set $\mathbf{U}_p$ to be the first p columns of $\mathbf{U}$
  Perform QRD on $\mathbf{U}_p^T$ and get $\mathbf{U}_p^T\mathbf{P}_p = \mathbf{QR}$
  Permutation matrix $\mathbf{P}_p$ identifies the p rows
end procedure

- **Input**: a specific layer which needs to be simplified
- Iteratively eliminates neurons based on QR factorization with column pivoting*
- **Output**: new DNN structures with updated edge weights which show a tradeoff in energy vs accuracy space

* Chan, Tony F. "Rank revealing QR factorizations." Linear algebra and its applications 88 (1987): 67-82.

Outline

- Challenges in realization of modern Deep Neural Networks (DNNs)
- Recent advances on efficient realization of DNNs
- Our contributions and approach
- Results and conclusions

Results: Information about Experimental DNNs

	LetNet5	LeNet300100	CIFAR10	CaffeNet
CS1	5×5×1×20	-	5×5×3×32	11×11×3×96
CS2	5×5×20×50	-	5×5×32×32	5×5×48×256
CS3	-	-	5×5×32×64	3×3×256×384
CS4	-	-	-	3×3×192×384
CS5	-	-	-	3×3×192×256
FC1	4×4×50×500	784×300	4×4×64×10	6×6×256×4096
FC2	500×10	300×100	-	4096×4096
FC3	-	100×10	-	4096×1000
#Edgs	2293K	266K	12.3M	724M
#Parm	431K	266K	89.4K	61M

- MNIST is a dataset of handwritten digits, contains 70000 28x28 gray-scale images (60000 training, 10000 testing) in 10 classes.
- CIFRAR10 dataset consists of 60000 32x32 color images (50,000 training and 10,000 testing) in 10 classes.
- ImageNet dataset consists of 1331167 256x256 color images (1281167 training and 50000 testing) in 1000 classes.

Results: Performed Experiments

- Applied our neuron elimination technique to various fully connected (FC) layers in each DNN and measured required memory as well as accuracy and energy tradeoffs.

- Accuracy is measured by running the testing images in corresponding dataset and measuring the classification error rate.

- Energy is measured by using the discussed energy model.

Results: Comparison of Required Memory

	Original		After				Accuracy	
	Params	Total	Params	Total	%original	Ratio	Original	After
LeNet5-FC1	13.8 Mb	13.9 Mb	1.1 Mb	1.2 Mb	8.13%	12.30X	99.10%	97.26%
LeNet5-FC2	13.8 Mb	14.0 Mb	1.7 Mb	1.9 Mb	12.67%	7.88X	99.10%	97.25%
LeNet300-100-FC1	8.5 Mb	8.5 Mb	1.6 Mb	1.6 Mb	18.85%	5.30X	98.21%	96.57%
LeNet300-100-FC2	8.5 Mb	8.5 Mb	1.5 Mb	1.5 Mb	17.64%	5.67X	98.21%	96.24%
LeNet300-100-FC3	8.5 Mb	8.5 Mb	7.6 Mb	7.6 Mb	89.40%	1.12X	98.21%	96.92%
CIFAR10-FC1	2.9 Mb	3.2 Mb	2.5 Mb	2.8 Mb	89.21%	1.12X	81.49%	79.57%
CaffeNet-FC1	243.8 MB	244.7 MB	129.1 MB	130.0 MB	52.94%	1.89X	56.67% / 79.59%	53.72% / 77.67%
CaffeNet-FC2	243.8 MB	244.7 MB	134.8 MB	135.3 MB	55.30%	1.81X	56.67% / 79.59%	54.19% / 77.80%
CaffeNet-FC3	243.8 MB	244.7 MB	186.2 MB	187.1 MB	76.37%	1.31X	56.67% / 79.59%	53.57% / 77.61%

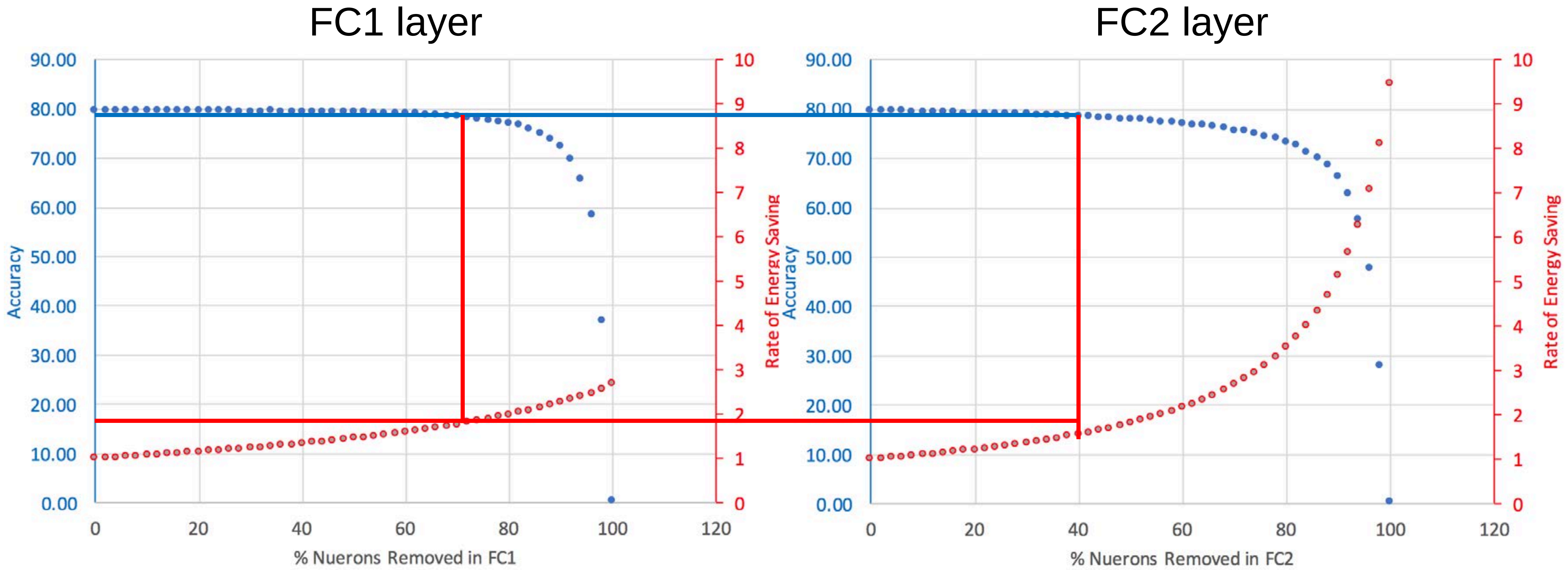
- Achieved significant memory saving with negligible loss in accuracy

Results: Energy Comparison (One Image Classification)

	Computation Energy	Communication Energy			Total	Accuracy
	MAC	$SRAM_{weights}$	$SRAM_{activations}$	DRAM		
LeNet5 (Original)	10.55 uJ	2.15 uJ	49.74 nJ	275.52 uJ	288.27 uJ	99.10%
LeNet5 (After)	1.95 uJ	0.05 uJ	38.05 nJ	6.69 uJ	8.73 uJ	97.26%
Original/After	5.42X	41.20X	1.31X	41.20X	33.02X	
LeNet300-100 (Original)	1.22 uJ	1.33 uJ	11.94 nJ	170.51 uJ	172.56 uJ	98.21%
LeNet300-100 (After)	0.05 uJ	0.05 uJ	2.09 nJ	6.36 uJ	6.46 uJ	96.24%
Original/After	26.81X	26.81X	5.71X	26.81X	26.71X	
CIFAR10 (Original)	56.57 uJ	0.45 uJ	0.14 uJ	57.24 uJ	114.40 uJ	81.94%
CIFAR10 (After)	43.50 uJ	0.40 uJ	0.13 uJ	51.07 uJ	95.10 uJ	79.57%
Original/After	1.30X	1.12X	1.08X	1.12X	1.20X	
CaffeNet (Original)	3.33 mJ	0.30 mJ	4.16uJ	39.01 mJ	42.64 mJ	56.67% / 79.59%
CaffeNet (After)	3.13 mJ	0.13 mJ	4.08uJ	16.07 mJ	19.33 mJ	53.72% / 77.67%
Original/After	1.06X	2.43X	1.02X	2.43X	2.21X	

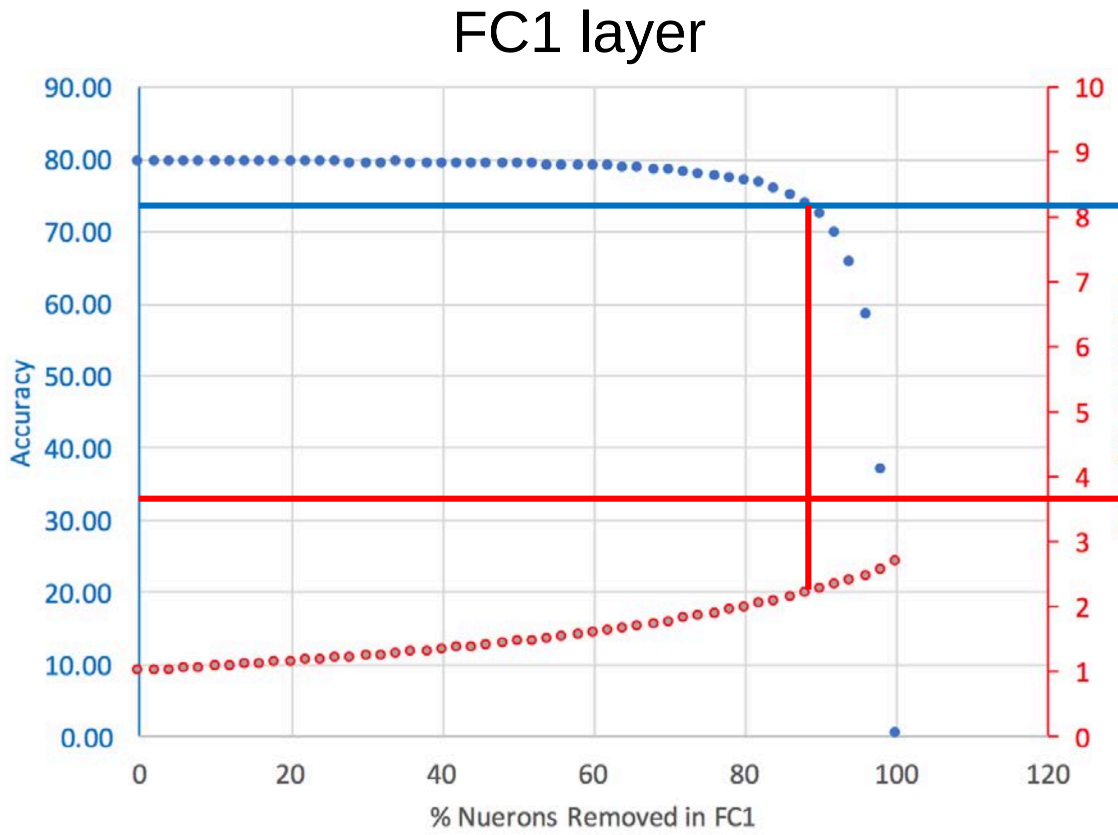
- Achieved significant energy saving with negligible loss in accuracy
- Most of the energy is consumed by DRAM

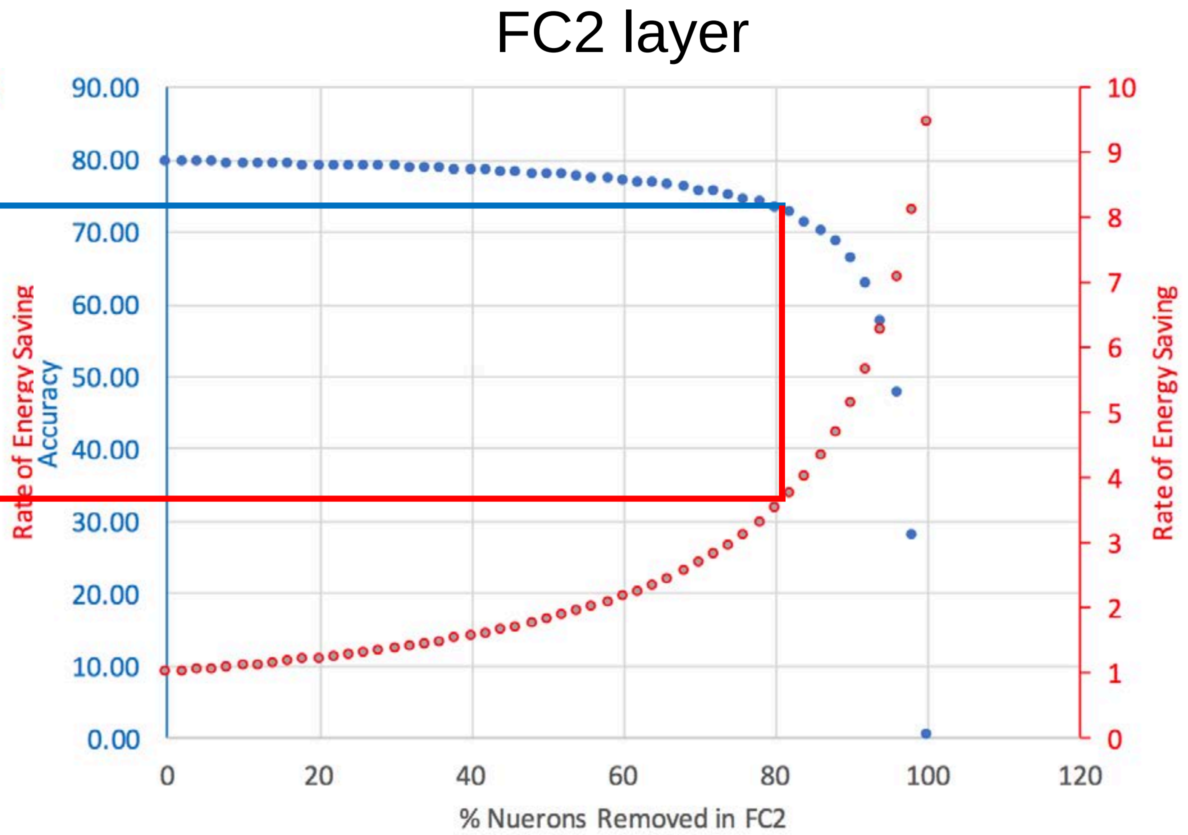
Results: Energy vs Accuracy Tradeoff in Each Layer



- Tradeoff in accuracy and energy when eliminating neurons in different layers of CaffeNet
- Example: for 79% accuracy we obtain higher rate of energy saving if FC1 is simplified

Results: Energy vs Accuracy Tradeoff in Each Layer





- Tradeoff in accuracy and energy when eliminating neurons in different layers of CaffeNet
- Example: for 73% accuracy we obtain higher rate of energy saving if FC2 is simplified

Conclusions

- Introduced a new neuron elimination technique which explicitly considers energy minimization as a design metric

- Showed the choice of layer to simplify in order to obtain maximum energy saving depends on the desired accuracy

Q & A

Question?


THANK
YOU
GRACIAS
ARIGATO
SHUKURIA
DANKSCHEEN
TASHAKKUR ATU
YAQHANYELAY
SUKSAMA
EKHMET
MEHRBANI
GRAZIE
BOLZÏN
MERCI
BİYAN
SHUKRIA
TINGKI
PALDIES
JUSPAXAR
GOZAIMASHITA
EFCHARISTO
KOMAPSUMNIDA
MAAKE
SPASSIBO
SNACHALHUYA
NUHUN
CHALTU
WABEEJA
MAITEKA
HUI
SPASIBO
DENKAUJA
HENACHALHYA
ATTO
MERASTAWHY
GAEJTHO
TAVTAPUCH
MEDAWAGSE
AGUYJE
FAKAAUE
LAH
SIKOMO
GUI
HATUR
MINMONCHAR