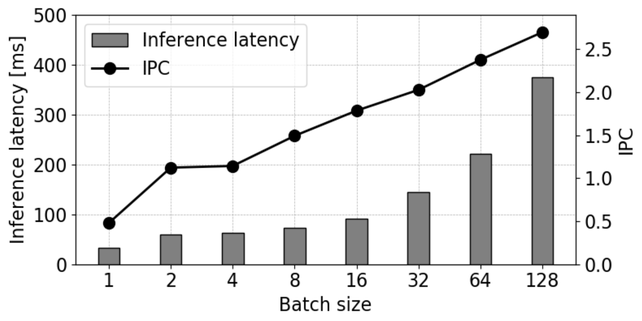
Deep Learning Inference on Commodity Network Interface Cards

Giuseppe Siracusano*, Davide Sanvito‡, Salvator Galea†, Roberto Bifulco*
*NEC Laboratories Europe, ‡Politecnico di Milano, †University of Cambridge

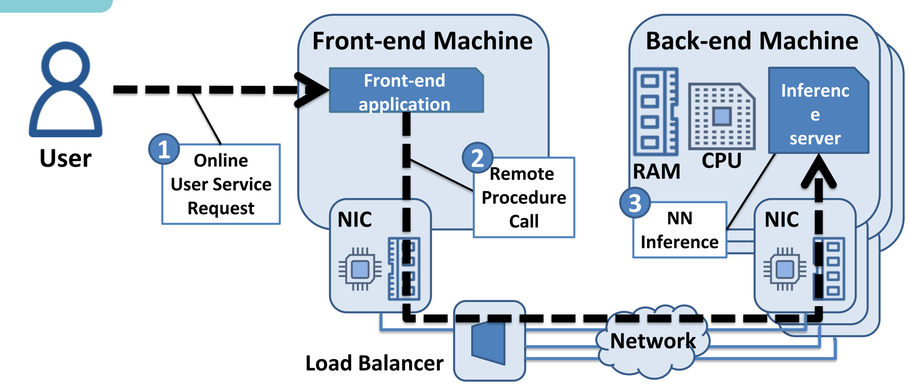
Problem:



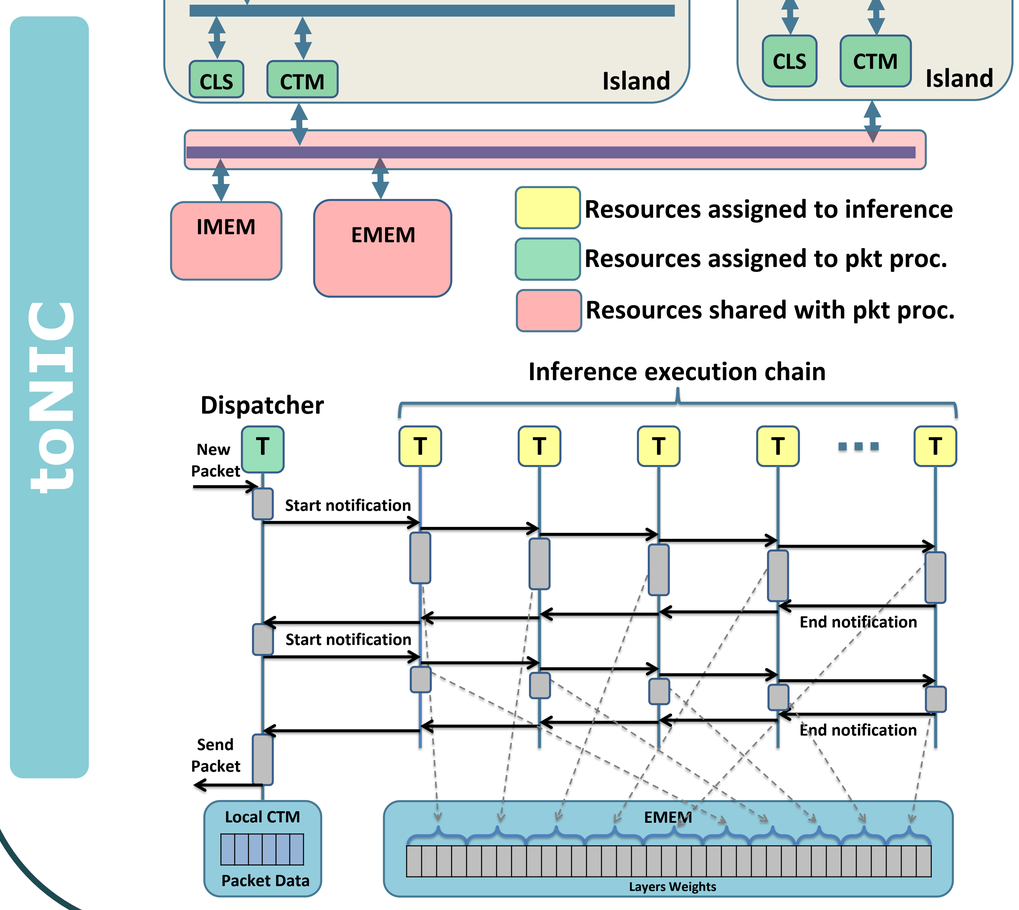
	MLP	Bin-MLP
Dataset	MNIST	
Layers	3 (bin)FC	
Acc.	98.7%	98.4%
	VGG	Bin-VGG
Dataset	CIFAR10	
Layers	13 conv, 3 (bin)FC	
Acc.	94.0%	94.0%

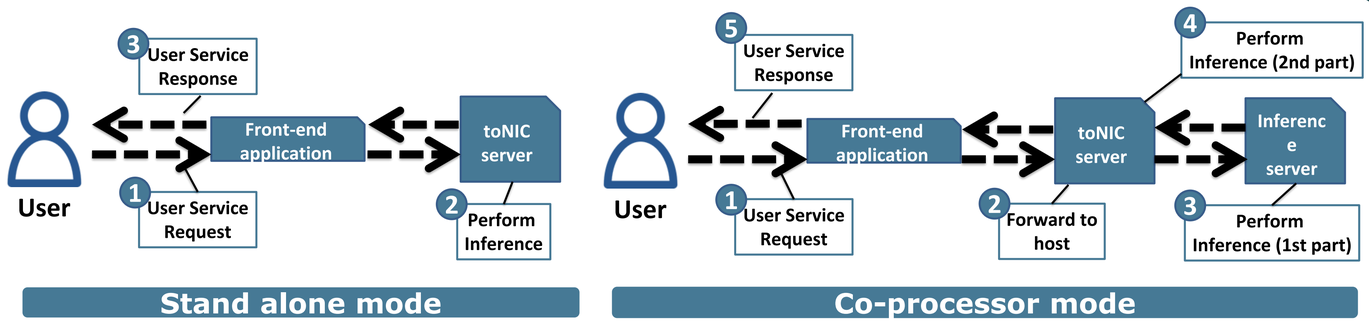
- Large share of neural network serving workloads is **memory-bound** (e.g., MLP over 60%[1])
- **Reduced efficiency on memory bound operations, i.e., low Instruction-per-cycle (IPC)**
- Batching improves IPC but increases latency
  - For time sensitive workload such increase is not tolerable [2]
- Model quantization helps
  - Extreme quantization, binarized models[3]
  - Matrix multiplications replaced with bitwise operations

Idea:



- Increase efficiency by reusing on-path network processors
  - SmartNICs and FPGAs frequently deployed inside datacenters [4]
  - A system-wide approach to NN serving!
- Network packet processing and NN memory-bound inference workloads may have complementary traits
  - Eg. per-packet parellel processing ≈ per-neuron parellel processing
- **In-network inference**





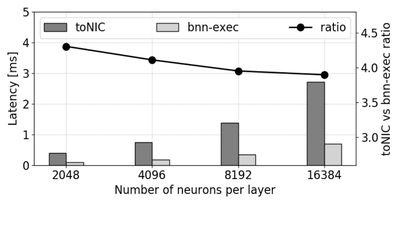
- **Can we do Neural Network inference on a commodity SmartNIC, while preserving high performance network communications?**

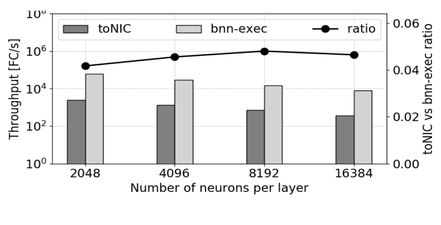
Yes, of course!!

- Reference SmartNIC architecture: Netronome NFP4000
  - Several processing cores (micro-engines, MEs), 8 threads per core
  - Heterogeneous memory structure
- Design:
  - Hardware resources are divided in two sets, pkt processing and NN inference
  - Packet processing threads have access priority on faster memories
  - NN inference threads can use a higher degree of parallelism

Results

	Power (vs Idle)	FC/s	FC/W vs Idle
Idle	69.4W	-	-
bnn-exec	145.9W (+75.5)	29520	386
toNIC	70.2W (+0.8)	1344	1680





- Packet forwarding (during inference):
  - line rate (80Gbps)
- Power efficiency, FC layers per Wat as a proxy for the cost of running the system
  - toNIC yields a 4.3x better performance/power ratio
- <u>But</u>: 3.7GHz CPU vs 800MHz NFP
  - Latency: ~4x higher in NFP (clock is 4.6x lower)
  - Throughput: ~5% of CPU throughput

What next?

- **What is the cost of modifying a SmartNIC to significantly improve its inference throughput?**
- Evaluated trough FPGA-based implementation
  - 256 neurons with 4096bit input values can be executed in parallel in only 80 us, using just 131KB of Block RAM.
  - 4096 x 4096 FC layer in only 1.3 ns, 781k FC/s using 2Mb of BRAM
  - Proposed design needs only 679 LUTs, less than 1% of the logic required to implement basic SmartNICs operations[5]
- A relatively small increase in the hardware resource requirements could improve NN processing throughput performance by a factor of 10-100

References:
[1] Norman P Jouppi, et al. "In-datacenter performance analysis of a tensor processing unit", ACM ISCA 2017.
[2] Ankit Singla, Balakrishnan Chandrasekaran, P Godfrey, and Bruce Maggs. "The internet at the speed of light", ACM HotNets 2014.
[3] Matthieu Courbariaux and Yoshua Bengio. "Binarynet: Training deep neural networks with weights and activations constrained to +1 or -1", CoRR.
[4] Firestone, Daniel, et al. "Azure Accelerated Networking: SmartNICs in the Public Cloud", USENIX NSDI 2018.
[5] Salvatore Pontarelli, et al. "Flowblaze: Stateful packet processing in hardware." USENIX NSDI 2019.

This work has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No. 761508 ("5GCITY"). This paper reflects only the authors' views and the European Commission is not responsible for any use that may be made of the information it contains.