Issues at the intersection of AI, Streaming, HPC, Data-centers and the Edge

NITRD Middleware and Grid Interagency Coordination Team (MAGIC)

Geoffrey Fox 1 April 2020

Digital Science Center, Indiana University

gcf@indiana.edu, http://www.dsc.soic.indiana.edu/

Some Simple Observations

- Consider **Science Research Benchmarks in MLPerf**
- Enhance **collaboration** between Industry and Research; HPC and MLPerf MLSys communities
- Support **common environments from Edge to Cloud and HPC systems**
- Huge switch to **Deep Learning for Big Data**
  - Many new algorithms to be developed
  - Deep Learning for (**Geospatial) Time Series** (staple of the edge) incredibly promising: obvious relevance to Covid-19 studies
- **Examples**
  - Inference at the edge
  - Fusion instabilities
  - Ride-hailing
  - Racing Cars
  - Images
  - Earthquakes
  - Solving ODE's
  - Particle Physics Events
- **Timely** versus **real-time** (throughput versus latency); both important

MLPerf Consortium Deep Learning Benchmarks

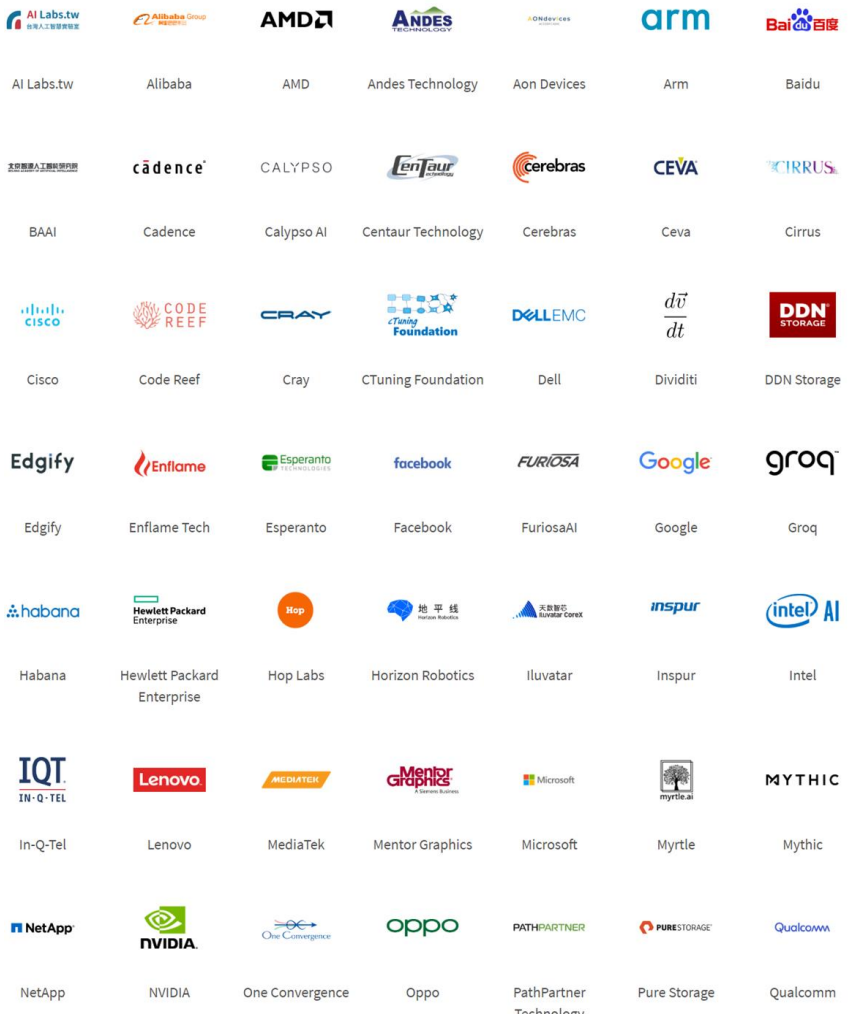
Some Relevant Working Groups

- Training
- Inference (Batch and Streaming)
- TinyML (embedded)
- Deep Learning for Time Series
- Power
- Datasets
- HPC (DoE Labs)
- Research
- Science Data
  (Proposed by Fox, Hey)

MLPerf's mission is to build fair and useful benchmarks for measuring training and inference performance of ML hardware, software, and services.

Benchmark what user sees

Companies



Researchers from

69 Companies; 8 universities

- Accelerate progress in ML via fair and useful measurement
- Serve both the commercial and research communities
- Enable fair comparison of competing systems yet encourage innovation to improve the state-of-the-art of ML
- Enforce replicability to ensure reliable results
- Keep benchmarking effort affordable so all can participate

MLPerf

Training v0.6

Typical Benchmarks
2048 TPUs or 1536 V100's with Infiniband quite powerful

Closed Division Times

#	Submitter	System	Processor	#	Accelerator	#	Software	Benchmark results (minutes)						
								Image classifi-cation	Object detection, light-weight	Object detection, heavy-wt.	Translation , recurrent	Translation , non-recur.	Recom-mendation	Reinforce-ment Learning
								ImageNet	COCO	COCO	WMT E-G	WMT E-G	MovieLens-20M	Go
								ResNet-50 v1.5	SSD w/ ResNet-34	Mask-R-CNN	NMT	Transformer	NCF	Mini Go
Available in cloud														
0.6-1	Google	TPUv3.32			TPUv3	16	TensorFlow, TPU 1.14.1.dev	42.19	12.61	107.03	12.25	10.20	[1]	
0.6-2	Google	TPUv3.128			TPUv3	64	TensorFlow, TPU 1.14.1.dev	11.22	3.89	57.46	4.62	3.85	[1]	
0.6-3	Google	TPUv3.256			TPUv3	128	TensorFlow, TPU 1.14.1.dev	6.86	2.76	35.60	3.53	2.81	[1]	
0.6-4	Google	TPUv3.512			TPUv3	256	TensorFlow, TPU 1.14.1.dev	3.85	1.79		2.51	1.58	[1]	
0.6-5	Google	TPUv3.1024			TPUv3	512	TensorFlow, TPU 1.14.1.dev	2.27	1.34		2.11	1.05	[1]	
0.6-6	Google	TPUv3.2048			TPUv3	1024	TensorFlow, TPU 1.14.1.dev	1.28	1.21			0.85	[1]	
Available on-premise														
0.6-7	Intel	32x 2S CLX 8260L	CLX 8260L	64			TensorFlow						[1]	14.43
0.6-8	NVIDIA	DGX-1			Tesla V100	8	MXNet, NGC19.05	115.22					[1]	
0.6-9	NVIDIA	DGX-1			Tesla V100	8	PyTorch, NGC19.05		22.36	207.48	20.55	20.34	[1]	
0.6-10	NVIDIA	DGX-1			Tesla V100	8	TensorFlow, NGC19.05						[1]	27.39
0.6-11	NVIDIA	3x DGX-1			Tesla V100	24	TensorFlow, NGC19.05						[1]	13.57
0.6-12	NVIDIA	24x DGX-1			Tesla V100	192	PyTorch, NGC19.05			22.03			[1]	
0.6-13	NVIDIA	30x DGX-1			Tesla V100	240	PyTorch, NGC19.05		2.67				[1]	
0.6-14	NVIDIA	48x DGX-1			Tesla V100	384	PyTorch, NGC19.05				1.99		[1]	
0.6-15	NVIDIA	60x DGX-1			Tesla V100	480	PyTorch, NGC19.05					2.05	[1]	
0.6-16	NVIDIA	130x DGX-1			Tesla V100	1040	MXNet, NGC19.05	1.69					[1]	
0.6-17	NVIDIA	DGX-2			Tesla V100	16	MXNet, NGC19.05	57.87					[1]	
0.6-18	NVIDIA	DGX-2			Tesla V100	16	PyTorch, NGC19.05		12.21	101.00	10.94	11.04	[1]	
0.6-19	NVIDIA	DGX-2H			Tesla V100	16	MXNet, NGC19.05	52.74					[1]	
0.6-20	NVIDIA	DGX-2H			Tesla V100	16	PyTorch, NGC19.05		11.41	95.20	9.87	9.80	[1]	
0.6-21	NVIDIA	4x DGX-2H			Tesla V100	64	PyTorch, NGC19.05		4.78	32.72			[1]	
0.6-22	NVIDIA	10x DGX-2H			Tesla V100	160	PyTorch, NGC19.05					2.41	[1]	
0.6-23	NVIDIA	12x DGX-2H			Tesla V100	192	PyTorch, NGC19.05			18.47			[1]	
0.6-24	NVIDIA	15x DGX-2H			Tesla V100	240	PyTorch, NGC19.05		2.56				[1]	
0.6-25	NVIDIA	16x DGX-2H			Tesla V100	256	PyTorch, NGC19.05				2.12		[1]	
0.6-26	NVIDIA	24x DGX-2H			Tesla V100	384	PyTorch, NGC19.05				1.80		[1]	
0.6-27	NVIDIA	30x DGX-2H, 8 chips each			Tesla V100	240	PyTorch, NGC19.05		2.23				[1]	
0.6-28	NVIDIA	30x DGX-2H			Tesla V100	480	PyTorch, NGC19.05					1.59	[1]	
0.6-29	NVIDIA	32x DGX-2H			Tesla V100	512	MXNet, NGC19.05	2.59					[1]	
0.6-30	NVIDIA	96x DGX-2H			Tesla V100	1536	MXNet, NGC19.05	1.33					[1]	

Benchmarks and rules

MLPerf

Area	Task	Model	Dataset	Quality	Server latency constraint	Multi-Stream latency constraint
Vision	Image classification	Resnet50-v1.5	ImageNet (224x224)	99% of FP32 (76.46%)	15 ms	50 ms
Vision	Image classification	MobileNets-v1 224	ImageNet (224x224)	98% of FP32 (71.68%)	10 ms	50 ms
Vision	Object detection	SSD-ResNet34	COCO (1200x1200)	99% of FP32 (0.20 mAP)	100 ms	66 ms
Vision	Object detection	SSD-MobileNets-v1	COCO (300x300)	99% of FP32 (0.22 mAP)	10 ms	50 ms
Language	Machine translation	GNMT	WMT16	99% of FP32 (23.9 BLEU)	250 ms	100 ms

Inference v0.5

Dividiti is MLPerf member in Cambridge UK

$$\frac{d\vec{v}}{dt}$$

Closed Division Times

			Benchmark results (Single Stream in milliseconds, MultiStream in no. streams, Server in QPS, Offline in inputs/second)									
			Image classification								Object detection	
			ImageNet				ImageNet				COCO	
			MobileNet-v1				ResNet-50 v1.5				SSD w/ MobileNet-v1	
ID	Submitter	System	Stream	MultiS	Server	Offline	Stream	MultiS	Server	Offline	Stream	MultiS
CATEGORY: Available												
Inf-0.5-1	Alibaba Cloud	Alibaba Cloud T4				17,473.60				5,540.10		
Inf-0.5-2	Dell EMC	Dell EMC R740			67,124.18	71,214.50			20,742.83	22,438.00		
Inf-0.5-3	Dell EMC	Dell EMC R740xd with 2nd generation Intel® Xeon® Scalable Processor									1.54	
Inf-0.5-4	Dell EMC	Dell EMC R740xd with 2nd generation Intel® Xeon® Scalable Processor									1.69	
Inf-0.5-5	dividiti	Raspberry Pi 4 (rpi4)	394.34				1,916.65					
Inf-0.5-6	dividiti	Raspberry Pi 4 (rpi4)	103.60				448.31					
Inf-0.5-7	dividiti	Linaro HiKey960 (hikey960)	121.11				518.07					
Inf-0.5-8	dividiti	Linaro HiKey960 (hikey960)	50.77				203.99					
Inf-0.5-9	dividiti	Linaro HiKey960 (hikey960)	143.07				494.90					
Inf-0.5-10	dividiti	Huawei Mate 10 Pro (mate10pro)	74.20				354.13					
Inf-0.5-11	dividiti	Huawei Mate 10 Pro (mate10pro)	111.60				494.92					
Inf-0.5-12	dividiti	Firefly-RK3399 (firefly)	120.56				695.11					
Inf-0.5-13	dividiti	Firefly-RK3399 (firefly)	106.49				447.90					
Inf-0.5-14	dividiti	Firefly-RK3399 (firefly)	80.12				391.02					
Inf-0.5-15	Google	Cloud TPU v3							16,014.29	32,716.00		
Inf-0.5-16	Google	2x Cloud TPU v3								65,431.40		
Inf-0.5-17	Google	4x Cloud TPU v3								130,833.00		
Inf-0.5-18	Google	8x Cloud TPU v3								261,587.00		
Inf-0.5-19	Google	16x Cloud TPU v3								524,978.00		
Inf-0.5-20	Google	32x Cloud TPU v3								1,038,510.00		
Inf-0.5-21	Habana Labs	HL-102-Goya PCI-board					0.24	700.00		14,451.00		
Inf-0.5-22	Intel	Intel® Xeon® Platinum 9200 processors									1.40	
Inf-0.5-23	Intel	Intel® Xeon® Platinum 9200 processors	0.49		27,244.81	29,203.30	1.37		4,850.62	5,965.62		
Inf-0.5-24	Intel	DELL ICL i3 1005G1	3.55			507.71	13.58			100.93	6.67	
Inf-0.5-25	NVIDIA	Supermicro 4029GP-TRT-OTO-28 8xT4 (T4x8)		6,320.00	135,073.00	141,807.00		1,920.00	41,546.64	44,977.80		2,624.00
Inf-0.5-26	NVIDIA	Supermicro 6049GP-TRT-OTO-29 20xT4 (T4x20)							103,532.10	113,592.00		
Inf-0.5-27	NVIDIA	SCAN 3XS DBP T496X2 Fluid (TitanRTXx4)		8,704.00	199,098.30	222,388.00		2,560.00	60,030.57	66,250.40		3,640.00
Inf-0.5-28	NVIDIA	NVIDIA Jetson AGX Xavier (Xavier)	0.58	302.00		6,520.75	2.04	100.00		2,158.93	1.50	102.00
Inf-0.5-29	Qualcomm	SDM855 QRD	3.02				8.95					

Science Data and MLPerf I

- Suggest that **MLPerf** should address **Science Research Data**
- There is **no existing scientific data benchmarking** activity with a similar flavor to MLPerf -- namely addressing important realistic problems aiming at modern data analytics including deep learning on modern high-performance analysis systems.
- Further, the challenges of science data benchmarking both benefit from the approach of MLPerf and will be synergistic with existing working groups.
- Science like industry involves **edge and data-center issues, inference, and training**, There are some similarities in the datasets and analytics as both industry and science involve **image data** but also differences;
  - Science data associated with **simulations** and particle physics experiments are quite different from most industry exemplars.
- Science datasets are often large and growing in size, while the multitude of active areas gives diverse challenges. **The best practice science algorithms are shifting to deep learning approaches** as in industry today.
- Benchmarks will help **more science fields** take advantage of modern ML and increase **link between Industry and Research**
- **Setting up first working group meeting:** Tell me if you are interested

Science Data and MLPerf II

- We foresee that scientific machine learning benchmarks for MLPerf will include a number of datasets, from each of the scientific domains, along with a representative problem from those domains. Some example benchmarks where we already have contacted scientists are:
  - Classifying cloud types from **satellite** imagery (**environmental sciences**)
  - Photometric redshift estimation based on observational data (**astronomy**), and
  - Removing noise from **microscopic** datasets (**life and material sciences**)
  - Real-time monitoring and archival analysis of data from **light sources** at DIAMOND (UK) and DoE Laboratories (US) (**Biological and Material sciences**)
  - **Simulations** covering near term recurrent neural networks and long term studies with fully connected and convolutional networks. This would initially be taken from **biomolecular and material science** areas but these examples will lead to work across many fields
  - Time series of **geographically distributed disease** occurrences with simulated and observed data
  - Monitoring of **plasma instabilities** in **fusion** Tokamaks with observation and simulation
- When fully contributed, the benchmark suite will cover the following domains: material sciences, environmental sciences, life sciences, fusion, particle physics, astronomy, earthquake and earth sciences, with more than one representative problem from each of these domains

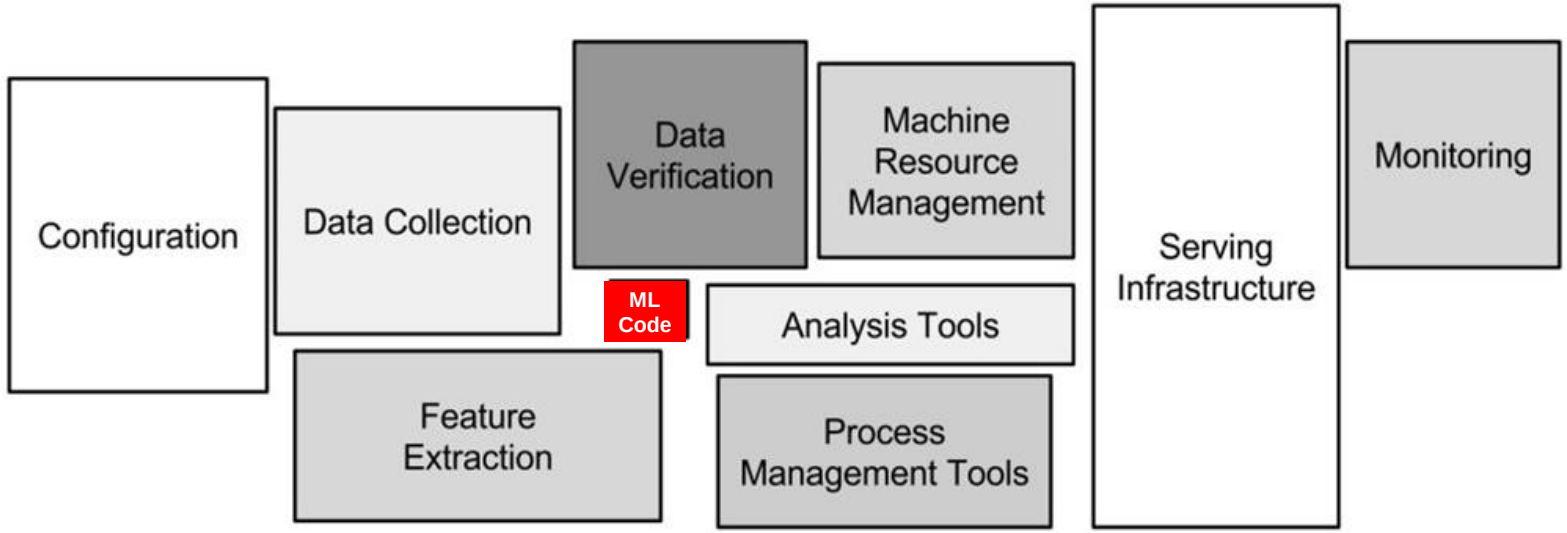
Linkage of Deep Learning (AI) and High Performance Computing

- On general principles, **Big data** stresses capabilities of compute/data platforms and needs best possible performance and **naturally uses HPC**
  - **HPC are not just Supercomputers** and the **use of HPC** for deep learning is pervasive in both industry and academia/government
  - Cloud/Supercomputer/HPC Cluster: GPU and TPU
  - **Edge:** FPGA Edge GPU, Edge TPU, Custom …
  - Large number of new architectures focussed on AI, CPU also useful!
- As well as **HPC for AI** (GPU's for deep learning), dramatic progress on **AI for HPC** enhancing simulations with 100's of papers in last 3 years
- **PyTorch** and **TensorFlow** (maybe MXNET) dominate; should collaborate on enhancing these and building systems around them
- **Hyper-parameter search needs to be deployed broadly;** places like IU do not have resources to support extensive hyper-parameter search but more common in Industry and DoE
- Need to advance **tools for time series**: LSTM, GRU, ConvLSTM, CNN+LSTM, Reformer, Transformer
  - Industry logistics, ride-hailing, speech, image streams but science can be different
- Need to advance **deep learning for clustering, dimension reduction** and other classic machine learning problems

HPCforML: Integration Challenges

Hidden Technical Debt in Machine Learning Systems

D. Sculley, Gary Holt, Daniel Golovin, Eugene Davydov, Todd Phillips
{dsculley,gholt,dgg,edavydov,toddphillips}@google.com
Google, Inc.



"Only a fraction of real-world ML systems is composed of ML code"

NIPS 2015 http://papers.nips.cc/paper/5656-hidden-technical-debt-in-machine-learning-systems.pdf

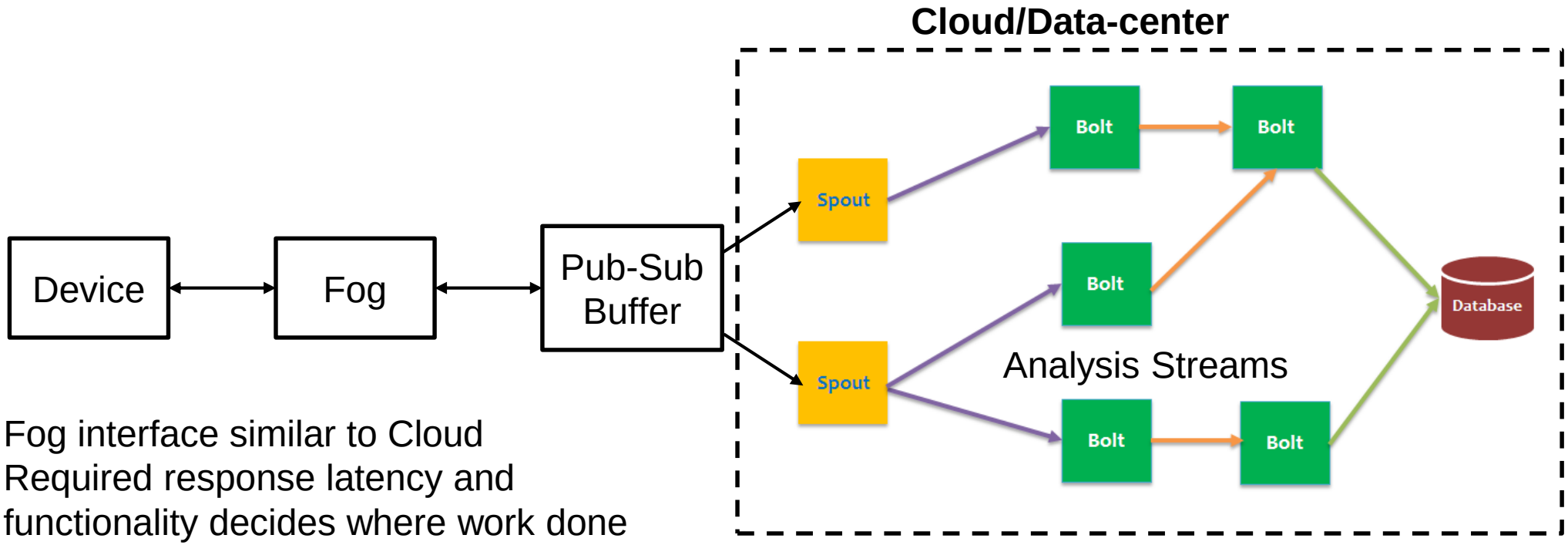
This well-known paper points out that parallel high-performance machine learning is perhaps most fun but just a part of system. We need to integrate in the other data and orchestration components.

This integration is not very good or easy partly because data management systems like Spark are JVM-based which doesn't cleanly link to C++, Python world of high-performance ML

My project Twister2 at IU addresses this problem

Not much addressed at MLSys

Classic Streaming Software Solution



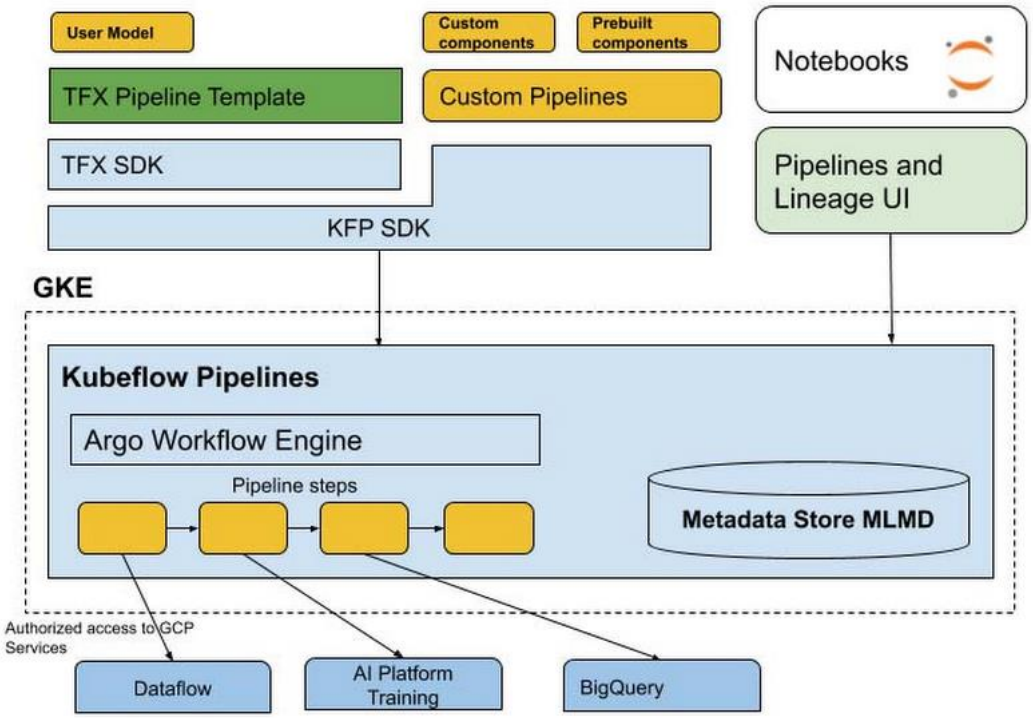
Fog interface similar to Cloud
Required response latency and functionality decides where work done

Apache Storm model supported by Apache Beam (Google Cloud Dataflow)

Streaming software surrounded by well known Big Data Engineering Systems such as Spark, Flink, Hadoop, MongoDB and use Serverless (Function as a Service) architecture
Java Python C++ integration not so easy with good performance

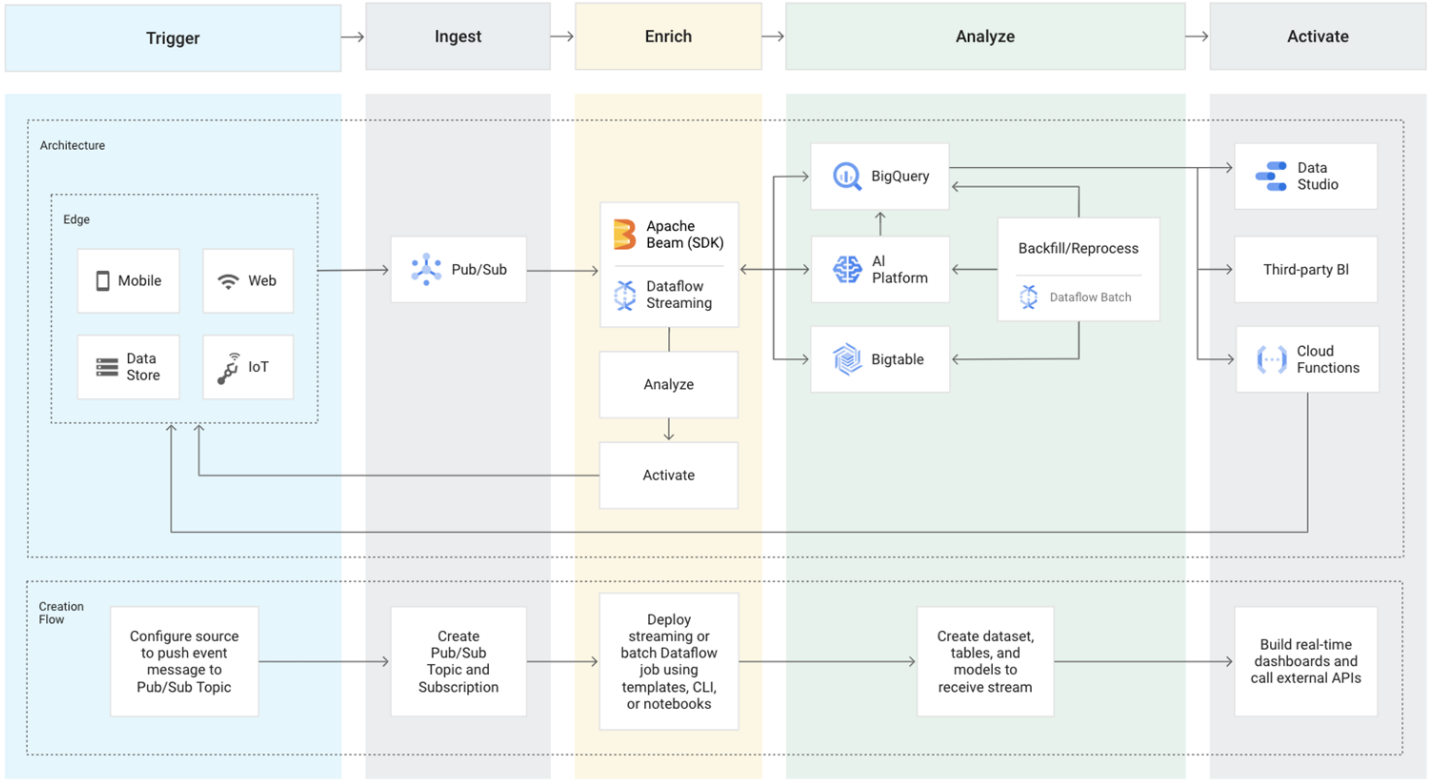
Apache Beam, Kubeflow, Argo: Typical Industry Edge to Cloud Workflow

- Recent Industry solution aimed at AI workflows constructed as a graph of containers
- March 11, 2020 https://cloud.google.com/blog/products/ai-machine-learning/introducing-cloud-ai-platform-pipelines



Google Cloud AI Platform Pipelines on Kubernetes
General DAG connected containers

Full edge to cloud workflow



MLSys Edge Deep Learning Papers on the Edge

- **Top conference at interface of Systems and AI**; >50% Industry
- *Federated Optimization in Heterogeneous Networks*; Distributed AI Loss functions
- *SkyNet: a Hardware-Efficient Method for Object Detection and Tracking on Embedded Systems*: Image analysis on edge
- *MNN: A Universal and Efficient Inference Engine*: Edge inference with CPU used in production by Alibaba (Strassen's algorithm)
- *Predictive Precompute with Recurrent Neural Networks*: DNN to predict user actions and suggest precomputing used in production by Facebook
- *Ordering Chaos: Memory-Aware Scheduling of Irregularly Wired Neural Networks for Edge Devices*: optimize scheduling
- *PoET-BiN: Power Efficient Tiny Binary Neurons*: Deep Learning on FPGA's and 4 other papers on edge quantization (low precision)

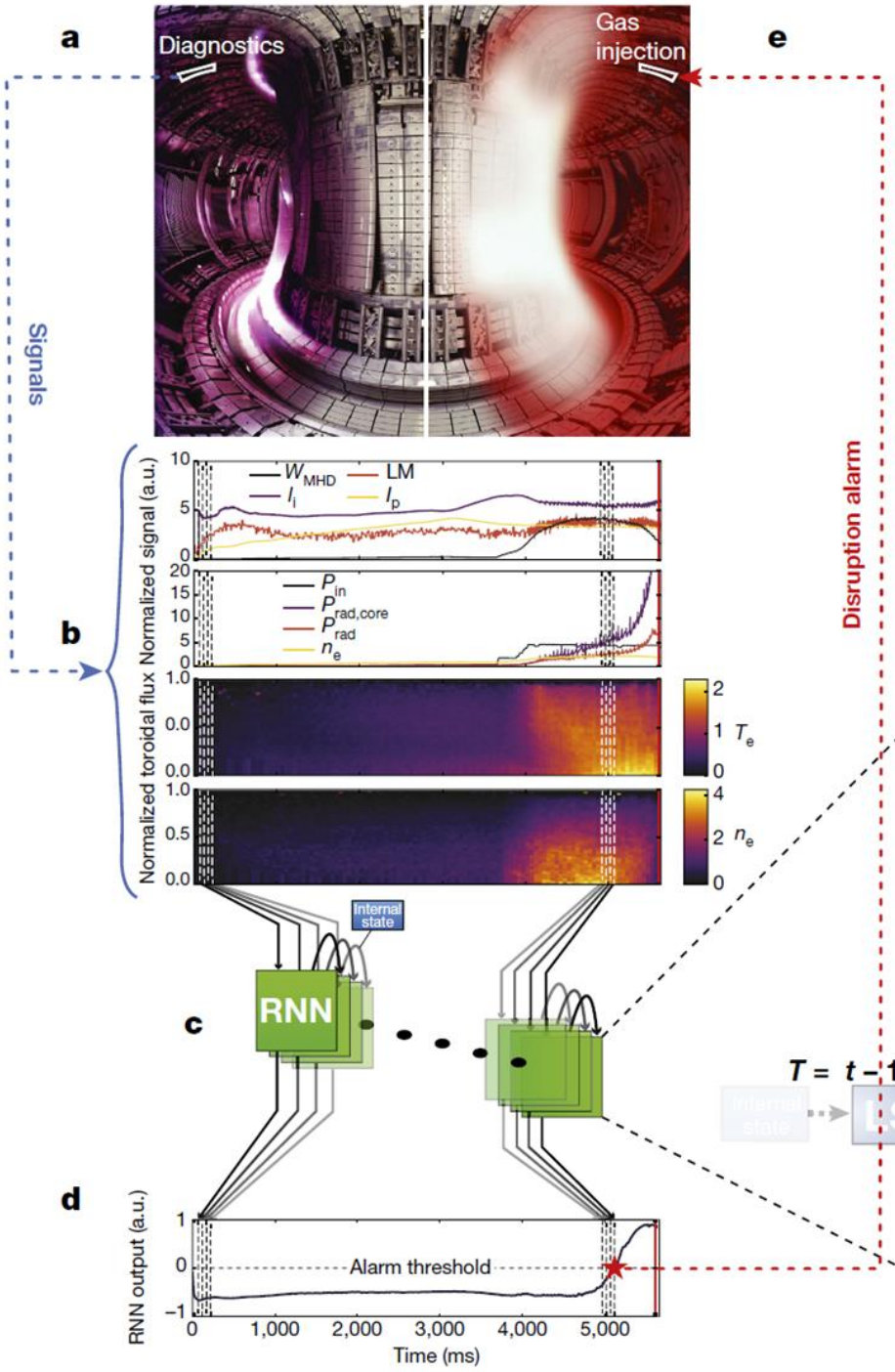
- Rest of papers (total 34 in main track) on server side and all focus on direct System-**Deep Learning** Integration

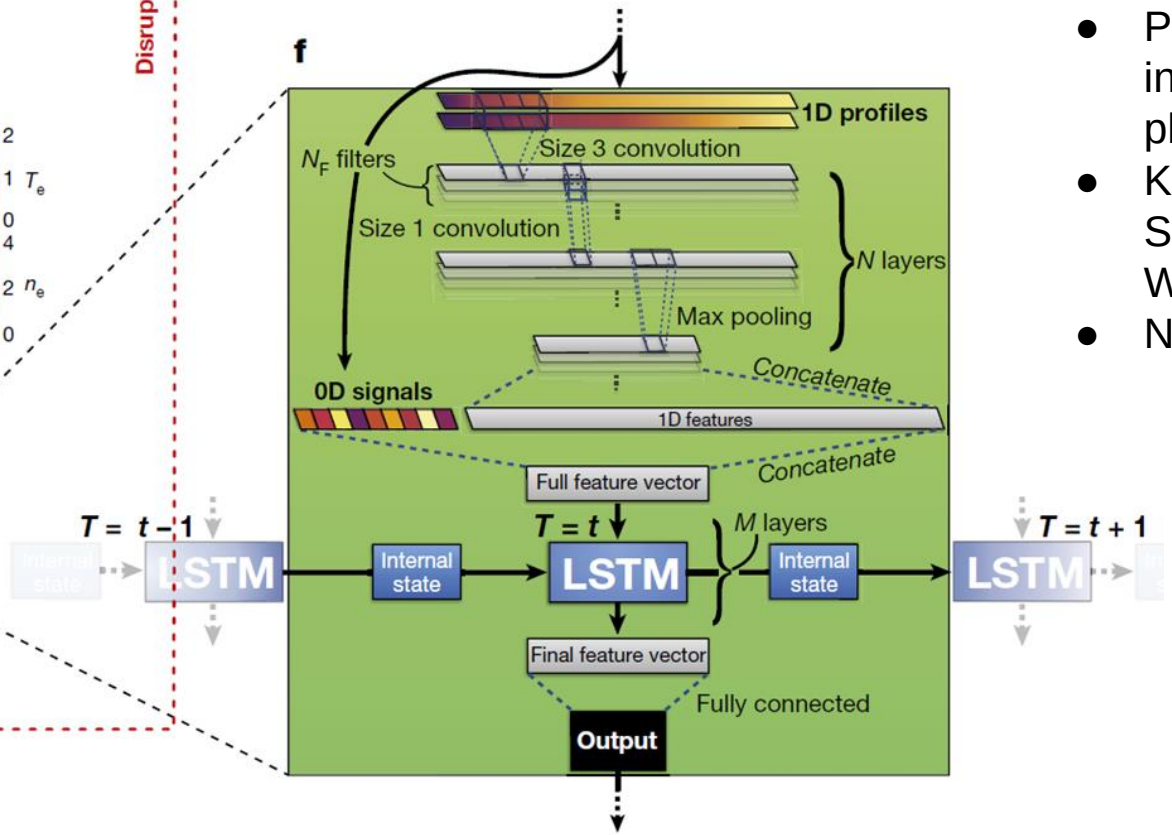
Deep Learning Prediction of Fusion Tokomak Instabilities

An example of Spatial Deep Learning

Streaming data at one per ms rate fed into CNN's followed by LSTM's

Creates "Disruption Alarm"

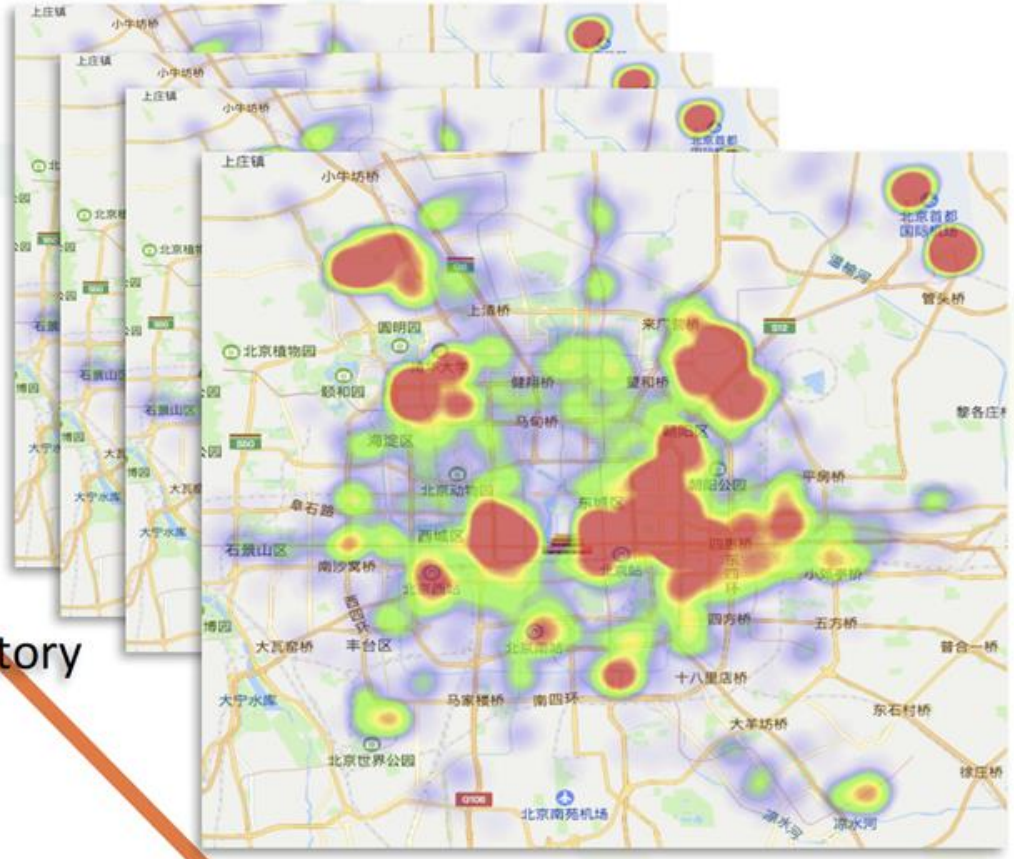




- Predicting disruptive instabilities in controlled fusion plasmas through deep learning
- Kates-Harbeck, Julian; Svyatkovskiy, Alexey; Tang, William
- Nature

http://arxiv.org/abs/1905.11395 X Geng, X Wu, L Zhang, Q Yang, Y Liu, J Ye, (Didi, HK, USC)
"Multi-Modal Graph Interaction for Multi-Graph Convolution Network in Urban Spatiotemporal Forecasting"
built on 3 different graphs (roads, function, geometry) plus RNN

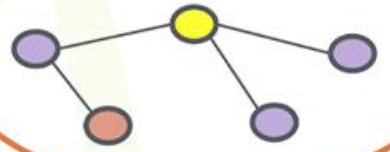
Apply to Ride-Hailing



Few minutes - hours-days-weeks

Spatio-temporal Forecasting

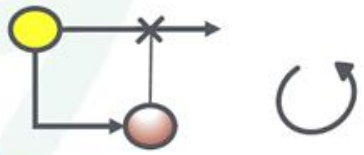
Graph Generation



Incorporating multi-relationship

Contextual-gated temporal modeling

Temporal: CGRNN



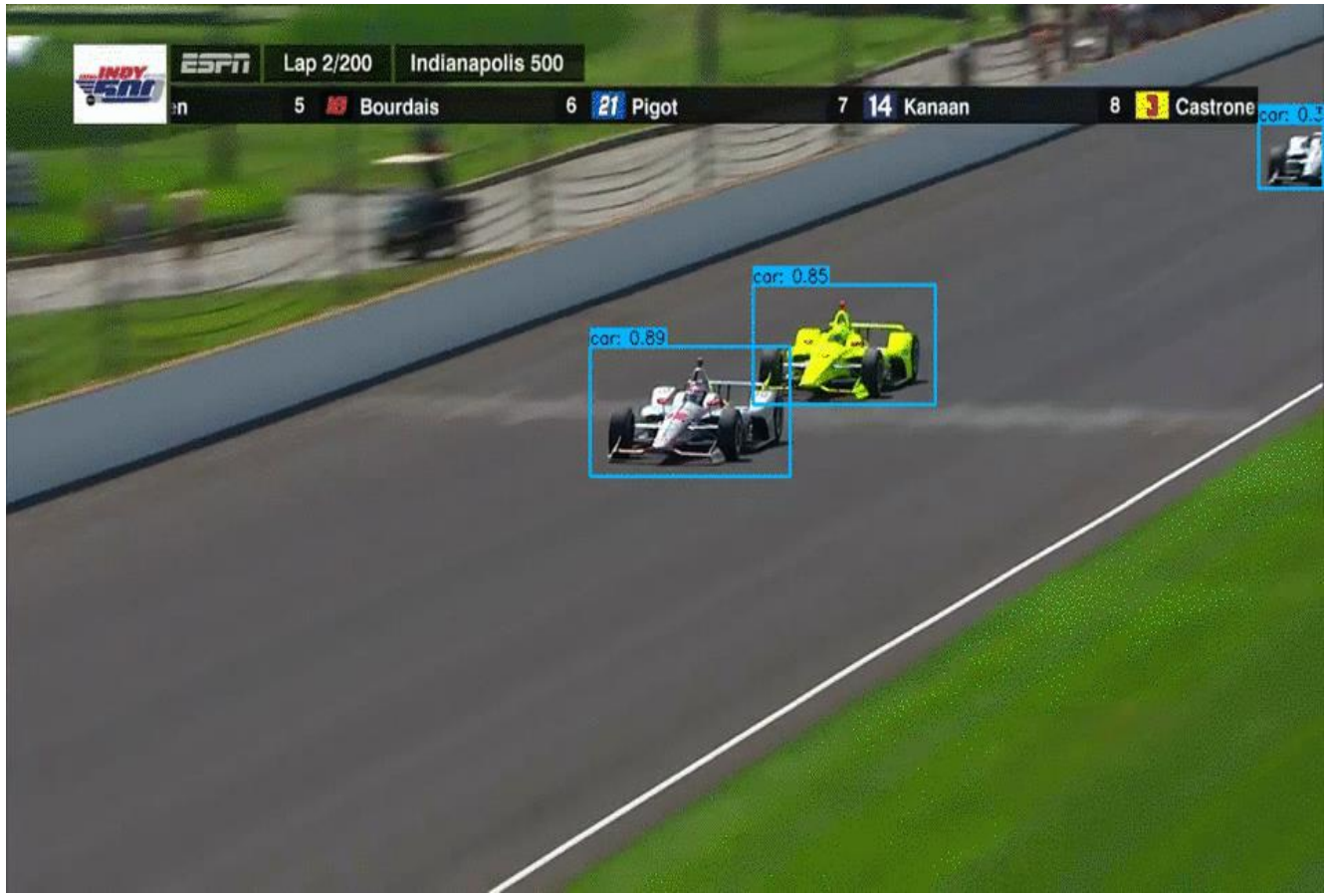
Spatio-MGCN

Non-euclidean spatial modeling

Multi-graph Convolutional Neural Network

Geng et al., AAAI 2019

Indianapolis 500 Real-Time Anomaly Detection and Ranking Prediction



Data Sources

Sensor data for IndyCar races.
Statistics of previous years: https://www.indycar.com/Stats

Selection of Features

Ranking prediction mainly depends on three measurable factors:

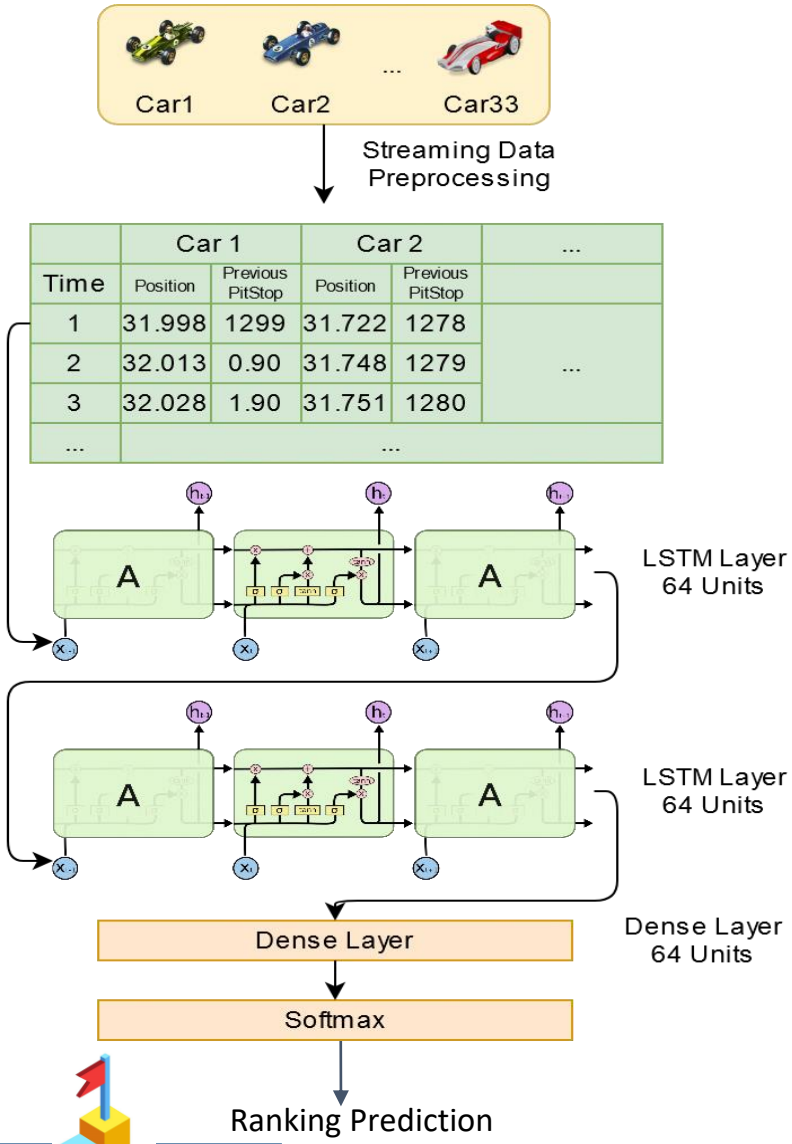
- **Past performance**: Time series data.
- **Current position:** The current Lap and Lap Distance.
- **Remaining fuel:** The time difference from the last Pit Stop.

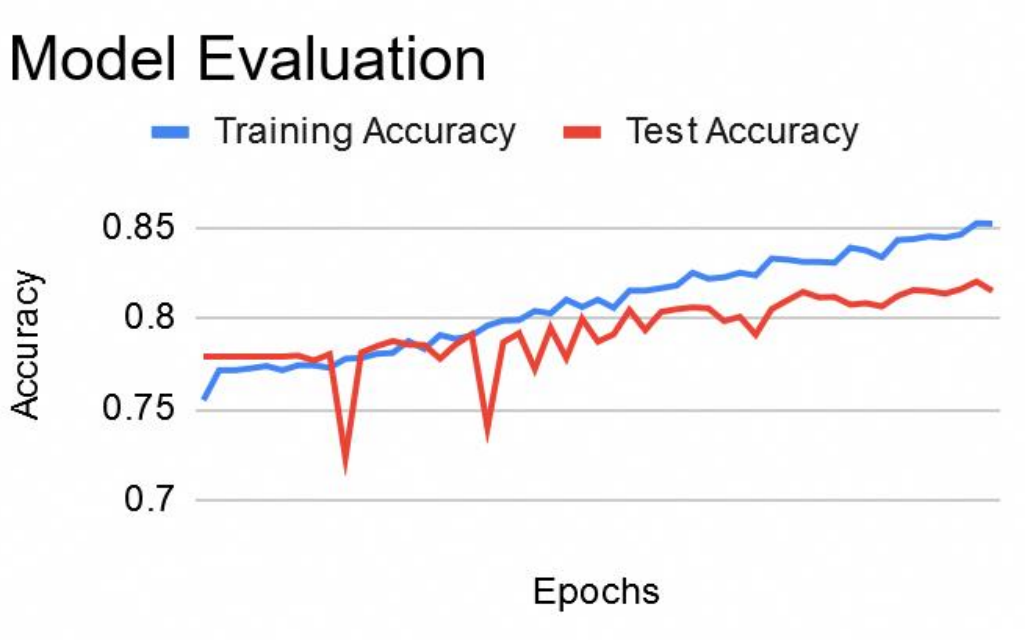
Data Preprocessing

Streaming data is adjusted to appropriate representation of time-series vector by interpolation methods.

Judy Qiu Indiana University and Jiayu Li in Research Group
Indy Car Race Analysis
Real-time video, track data, car sensor data

Indy500 Rank Prediction using LSTM on Streaming Data





Features Used

Position: The current position of the car.
Previous Pitstop: How long has passed since the last pit stop.

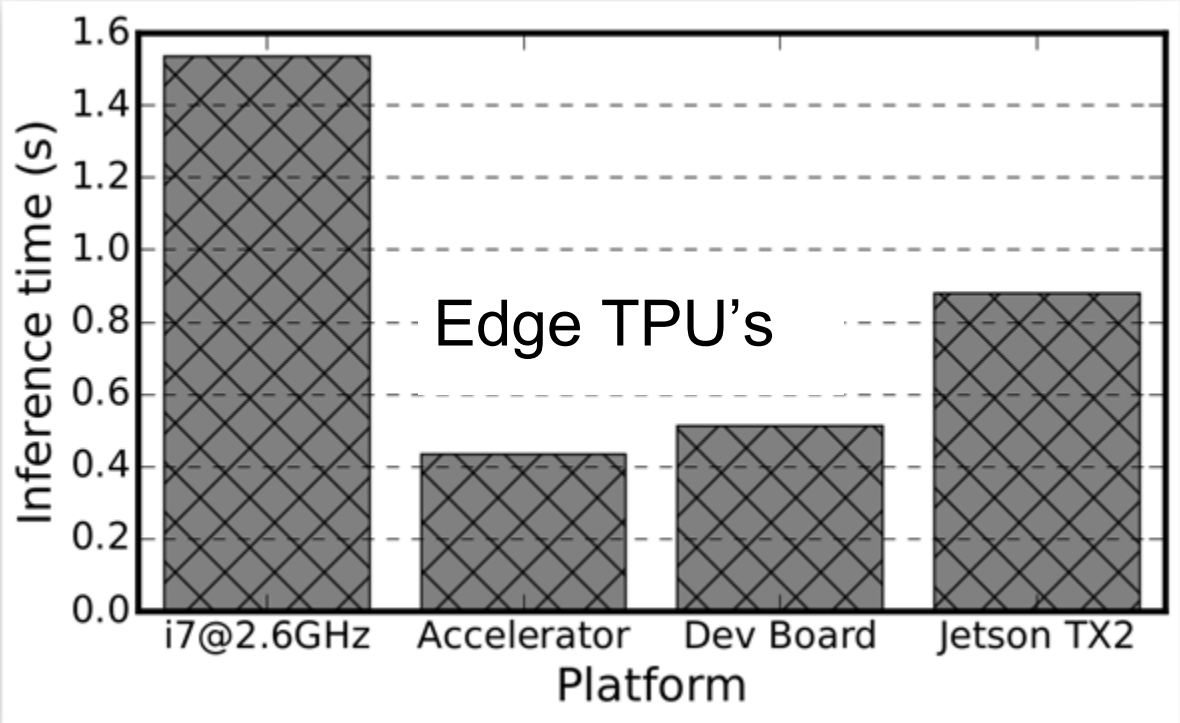
Dimensions of the Data Sets

A total of 66 features: 33 cars * 2 features each.
Each sample lasts approximately 10,000 seconds

Model Output

Probability distribution of the leading car.

Denoising Images from Light Sources (Argonne, IU)



Paper from SC20 Workshop

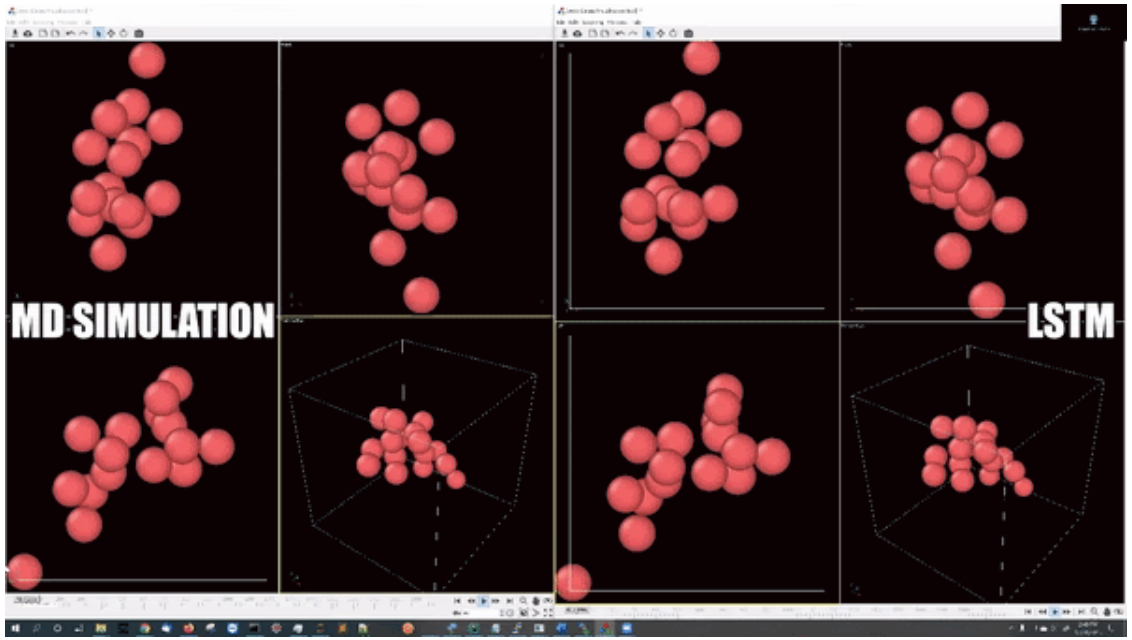
Scientific Image Restoration Anywhere

Vibhatha Abeykoon, Zhengchun Liu, Rajkumar Kettimuthu, Geoffrey Fox and Ian Foster

1024 by 1024 Image

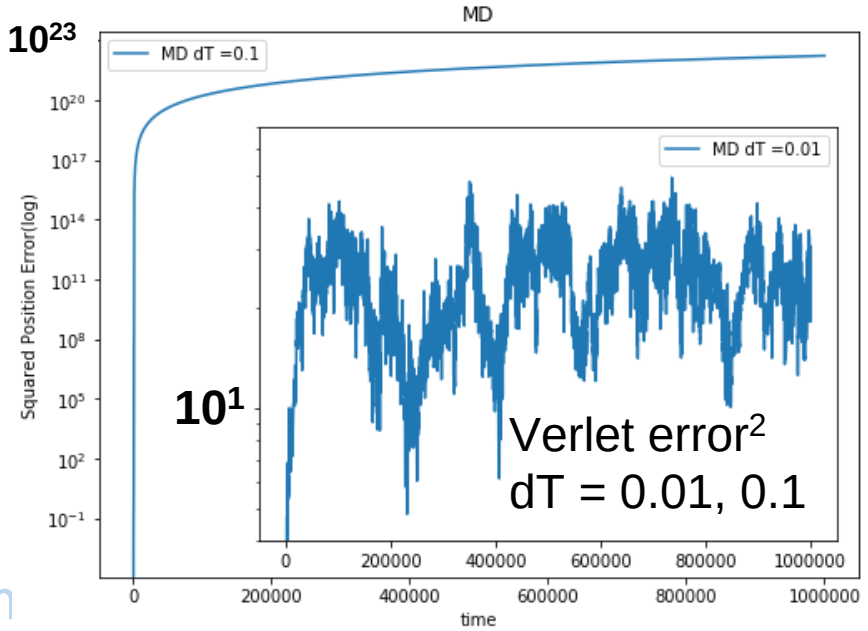
Learn Newton's laws with Recurrent Neural Networks

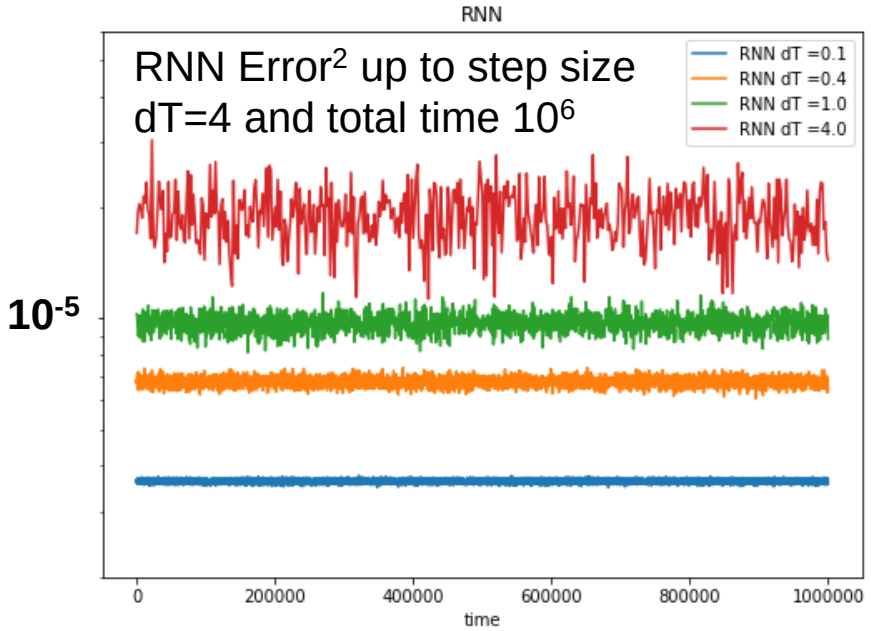
- **(work with JCS Kadupitiya, Vikram Jadhao)**
- **Deep Learning is revolutionizing (spatial) Time series Analysis**
- Good example is integrating sets of differential equations
- Train the network on traditional 5 time step series from (Verlet) difference equations
- Verlet needs time step .001 for reliable integration but
- Learnt LSTM network is reliable for time steps which are **4000 times longer** and also learn potential.



Speedup is 30000 on 16 particles interacting with Lennard-Jones potentials

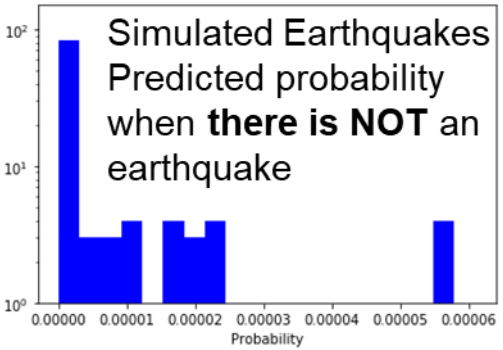
- 2 layer-64 units per layer LSTM network: **65,072 trainable parameters**
- 5000 training simulations



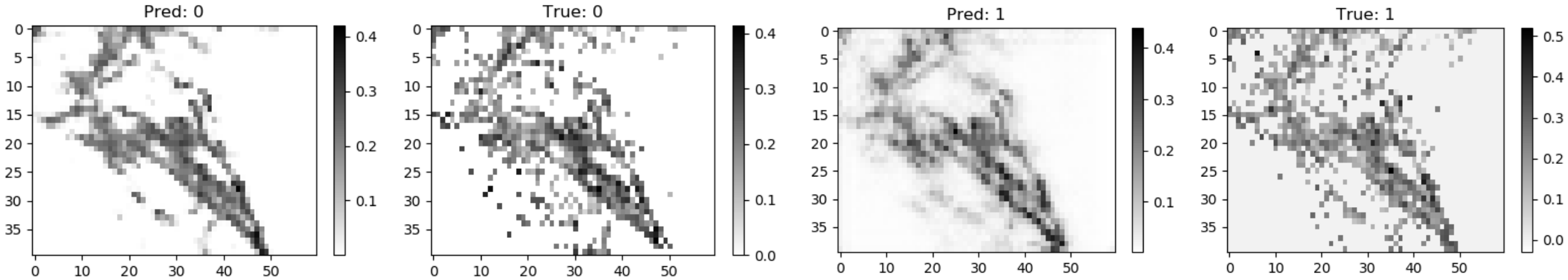


Hidden Theories and Hidden Instances

- The LSTM description of particle dynamics suggests that the DNN has learnt “Newton’s laws” and then you look at different instances in the inference
- To the right is model learning simulated earthquakes
- Below are two predictions compared to observation for annual earthquake activity in Southern California (each pixel is about 11 km square)
- Uses Convolutional LSTM with 5 time steps and data is log of aggregated energy released (can’t use energy as too large a dynamic range)



DNN learns theories

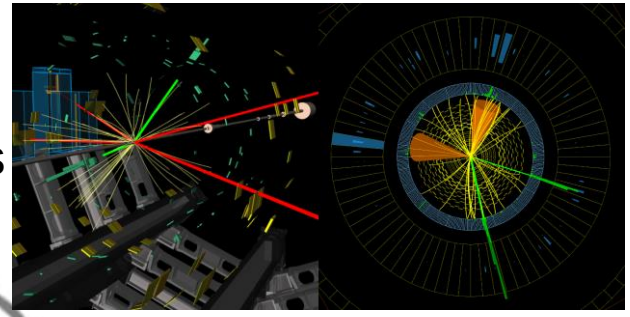


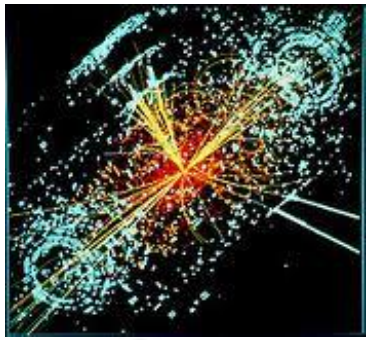
Deep Learning in Particle Physics Data Analysis

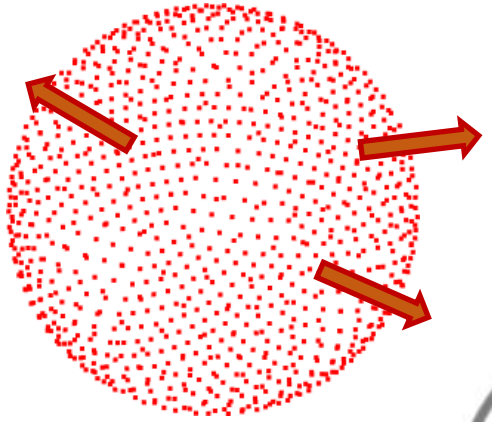
- Train on LHC and simulation events with input as angular distribution of momentum in a 4π steradian detector i.e. you have total energy transmitted in each direction on sphere surrounding interaction point

Different physics gives different patterns of particles

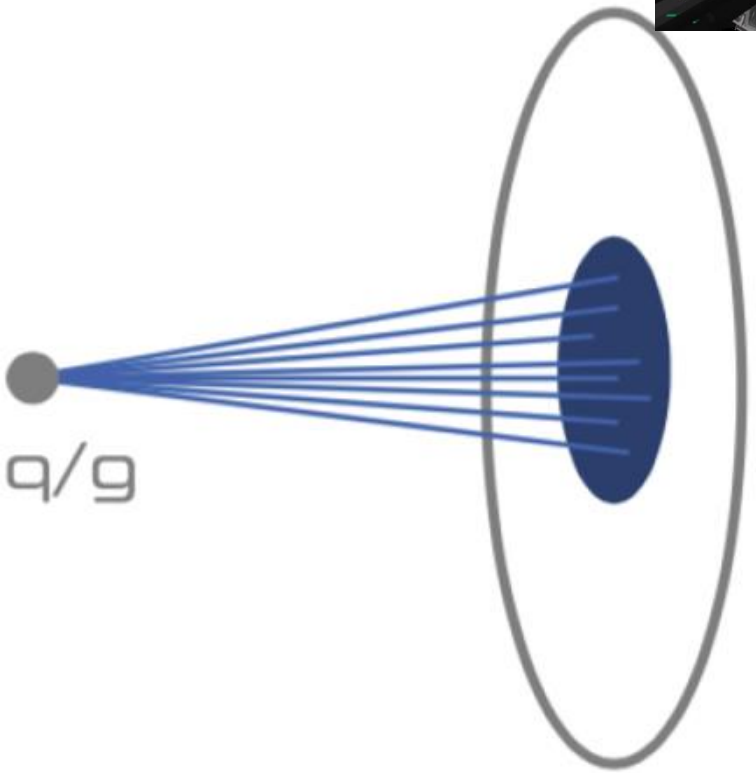
LHC Events

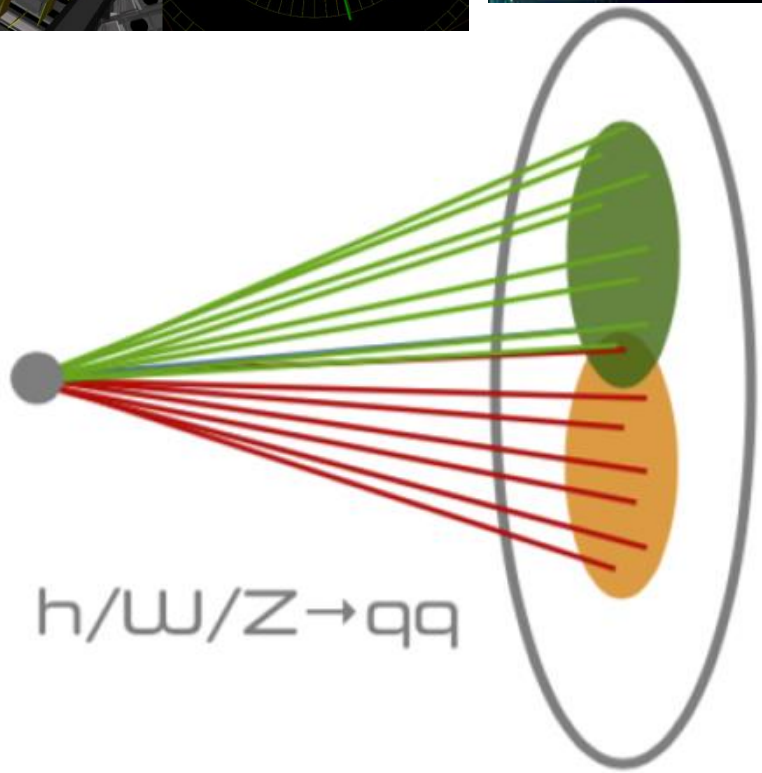


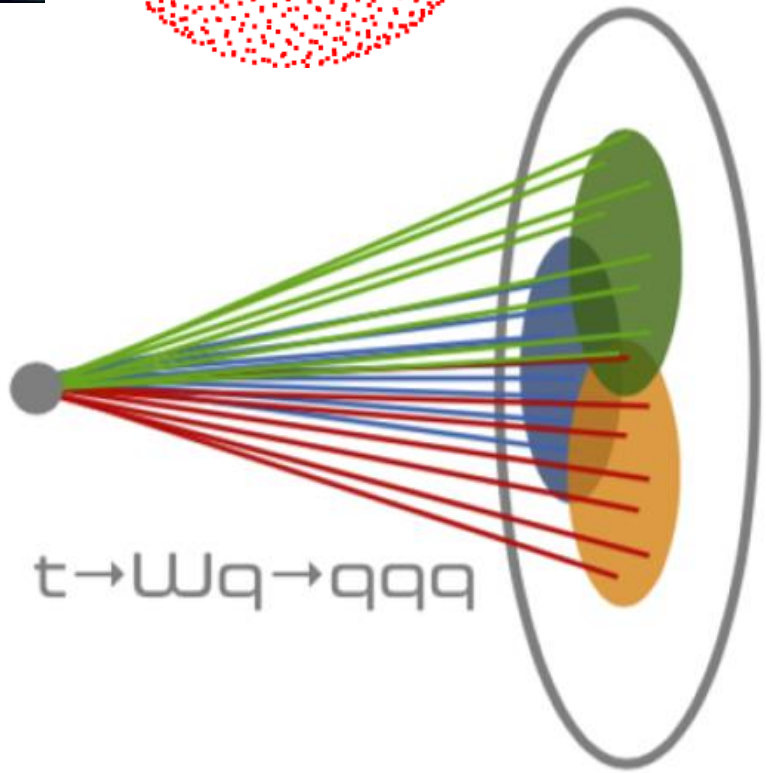




Energy flowing in each direction







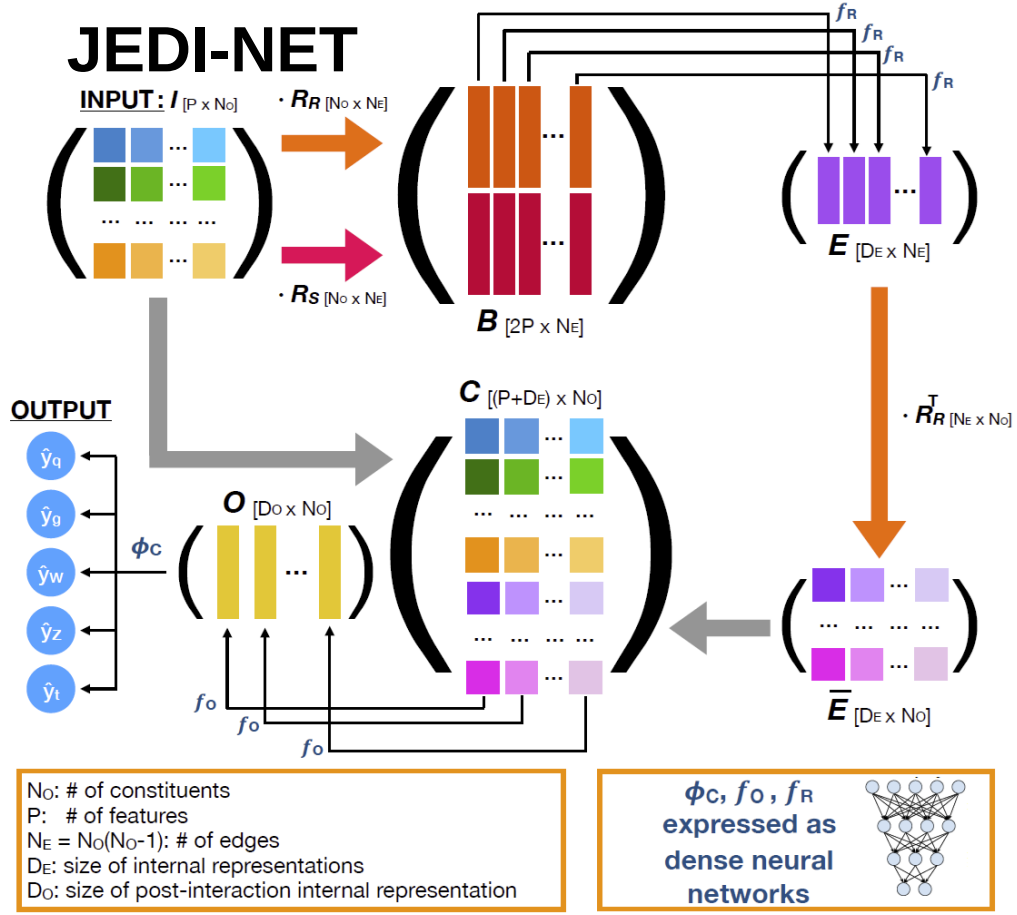
Deep Learning in Particle Physics Data Analysis

- (Caltech) *Observables for the analysis of event shapes in e+ e− annihilation and other processes*, GC Fox, S Wolfram, Physical Review Letters **1978, 1648 citations (50 in 2019)** introduced quantities to characterize shapes of collections of particles. They were invariant exactly to rotations and approximately to unknown details of decays of hidden particles (quarks, gluons, Higgs, W/Z bosons) as involved sums over momenta preserved in decays
  - Need tiny computing!
- (Caltech, Fermilab, CERN) arXiv:1908.05318 from CMS introduces JEDI-NET with 3 DNN's for this
- This just one of many classic ideas replaced by deep learning.

$$H_\ell = \sum_{i,j=1}^{N} \frac{|\vec{p}_i|}{\sqrt{s}} \frac{|\vec{p}_j|}{\sqrt{s}} \frac{4\pi}{2\ell+1} \sum_{m=-\ell}^{\ell} Y_\ell^m(\Omega_i) Y_\ell^{m*}(\Omega_j)$$

$$= \sum_{i,j=1}^{N} \frac{|\vec{p}_i||\vec{p}_j|}{s} P_\ell(\cos\Omega_{ij}) \,, \qquad \textbf{Fox Wolfram Moments}$$

$$\cos\Omega_{ij} = \cos\theta_i \cos\theta_j + \sin\theta_i \sin\theta_j \cos(\phi_i - \phi_j).$$



Conclusions: Reiterate Simple Observations

- Consider **Science Research Benchmarks in MLPerf**
- Enhance **collaboration** between Industry and Research; HPC and MLPerf/MLSys communities
- Support **common environments from Edge to Cloud and HPC systems**
- Huge switch to **Deep Learning for Big Data**
  - Many new algorithms to be developed
  - Deep Learning for (**Geospatial) Time Series** (staple of the edge) incredibly promising: obvious relevance to Covid-19 studies
- **Examples**
  - Inference at the edge
  - Fusion instabilities
  - Ride-hailing
  - Indy car racing
  - Images
  - Earthquakes
  - Solving ODE's
  - Particle Physics Events
- **Timely** versus **real-time** (throughput versus latency); both important

"Any opinions, findings, conclusions or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the views of the Networking and Information Technology Research and Development Program."

The Networking and Information Technology Research and Development (NITRD) Program

Mailing Address: NCO/NITRD, 2415 Eisenhower Avenue, Alexandria, VA 22314

Physical Address: 490 L'Enfant Plaza SW, Suite 8001, Washington, DC 20024, USA Tel: 202-459-9674, Fax: 202-459-9673, Email: nco@nitrd.gov, Website: https://www.nitrd.gov