HPC for AI at Facebook

Antoine Bordes – FAIR

High Performance Data Analytics Forum – CNRS – Oct. 17 2016

Key Figures

Every day on Facebook/WhatsApp:

- 60 billion text messages are sent
- 2 billion pictures are uploaded
- Several millions new videos are published
- 1.5 billion searches are conducted

Facebook AI Research

History

- Established in Dec 2013
- Initiative of CEO and CTOL
- Led by Yann Lecun



- Toward Artificial Intelligence (AI) with Machine Learning

Mission

Advance the state-of-the-art of AI

- Produce software tools for AI research and applications
- Help FB products to leverage advances in AI
  - Software prototyping, architecting, interaction with product teams…
- Contribute to Facebook IP portfolio
- Publish research in best conferences and journals



Today

~45 researcher scientists

Machine Learning, Natural Language Processing, Computer Vision, …

~25 research engineers

Software support, prototyping, in

3 locations: NYC, MPK, **Paris**

AI Research

Artificial Intelligence?

Design systems that perceive and reason about the environment to perform "human tasks"

Language: speech recognition, language translation, question answering, dialog system, etc.

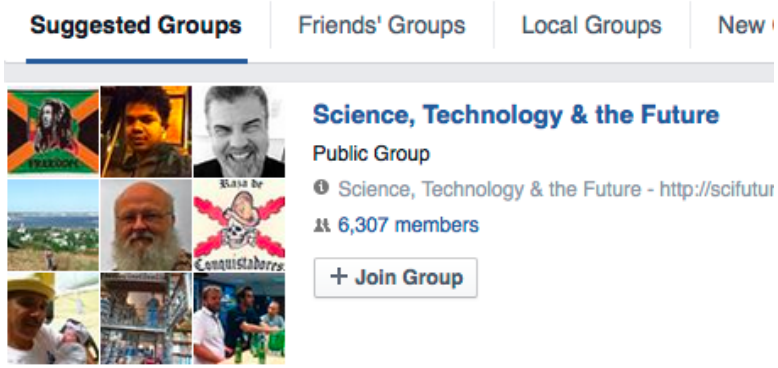
Vision: face detection/recognition, object/text recognition, action classification, natural language description, etc.

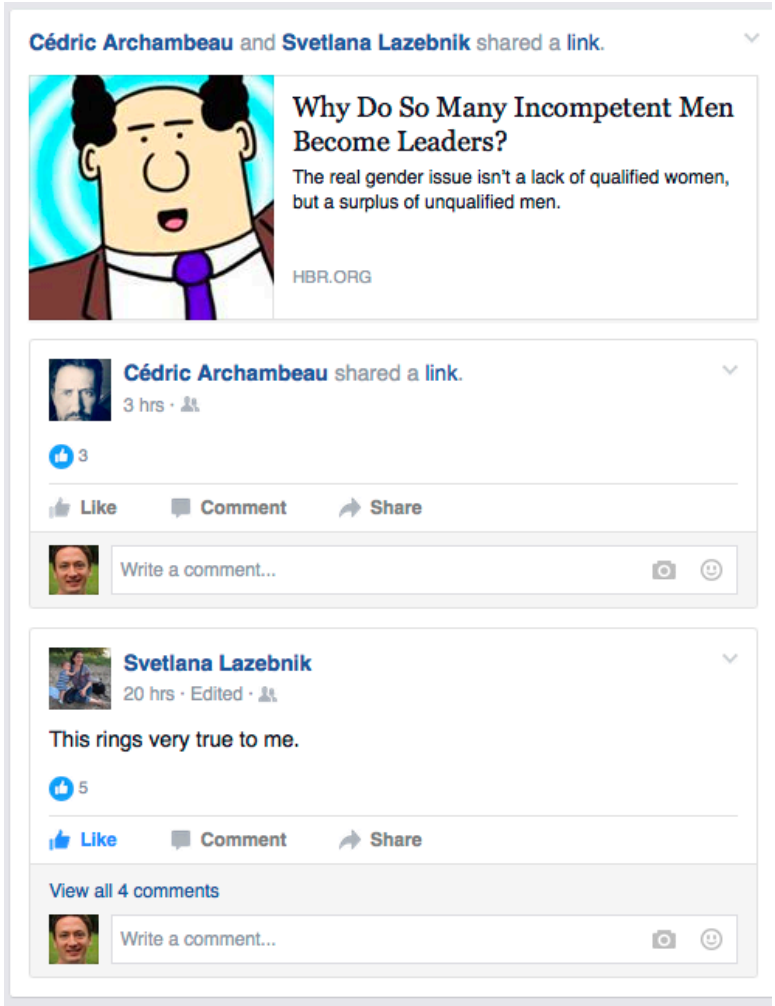
Planning: given starting point and end goal, plan strategy

FAIR Vision

AI will mediate communication:

- between people, e.g.:
  - feed ranking
  - friends / groups suggestions
  - real-time translation

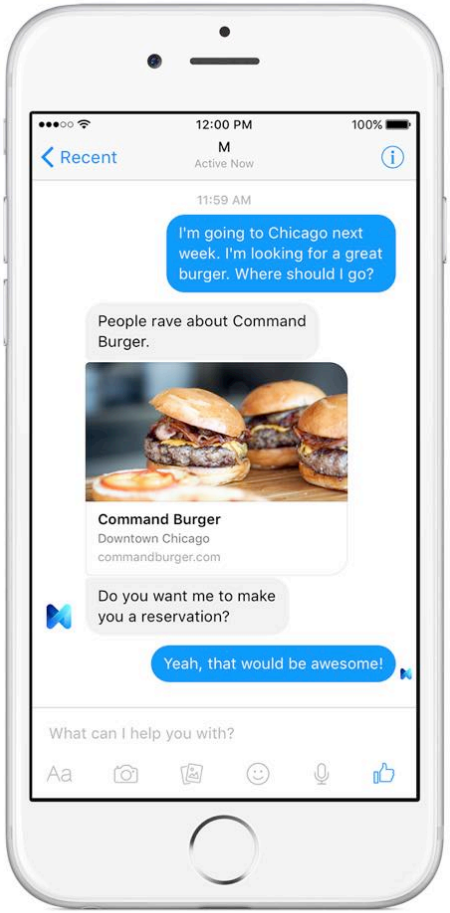




FAIR Vision

AI will mediate communication:

- between people, e.g.:
  - feed ranking
  - friends / groups suggestions
  - real-time translation
- between people and the digital world, e.g.:
  - content search
  - question answering
  - real-time dialog

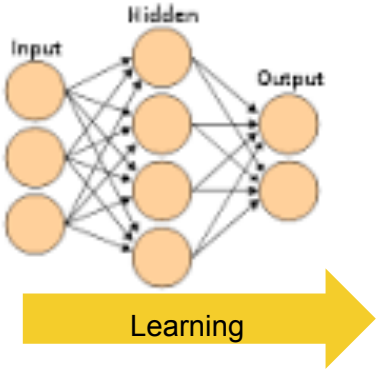


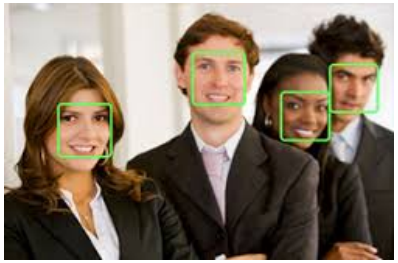
Machine Learning?

Learn complex functions from data to make future predictions

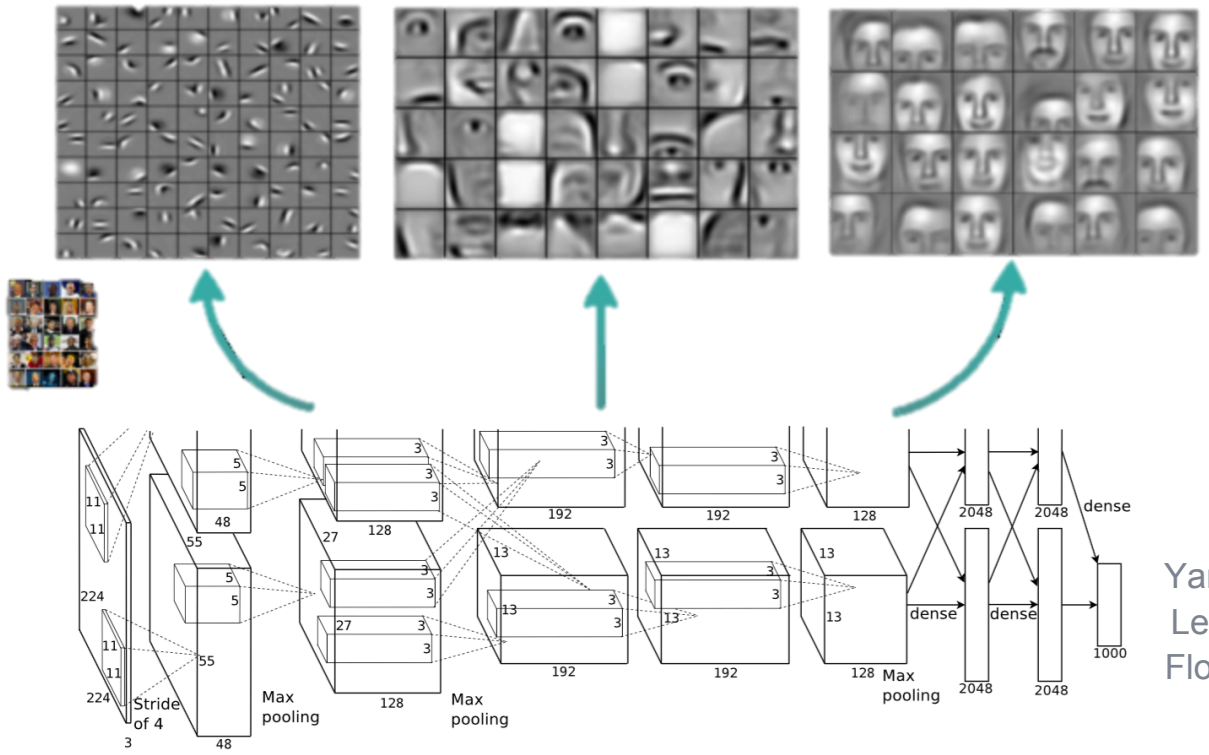
- we do not teach computers how to solve a problem
- we teach computers how to learn to solve a problem from data

  → Teaching done via optimizing parameters (millions/billions of them).





Deep Learning?



Yann (90%)
Leon (5%)
Florent (0.5%)

➔ learn hierarchical representations of data

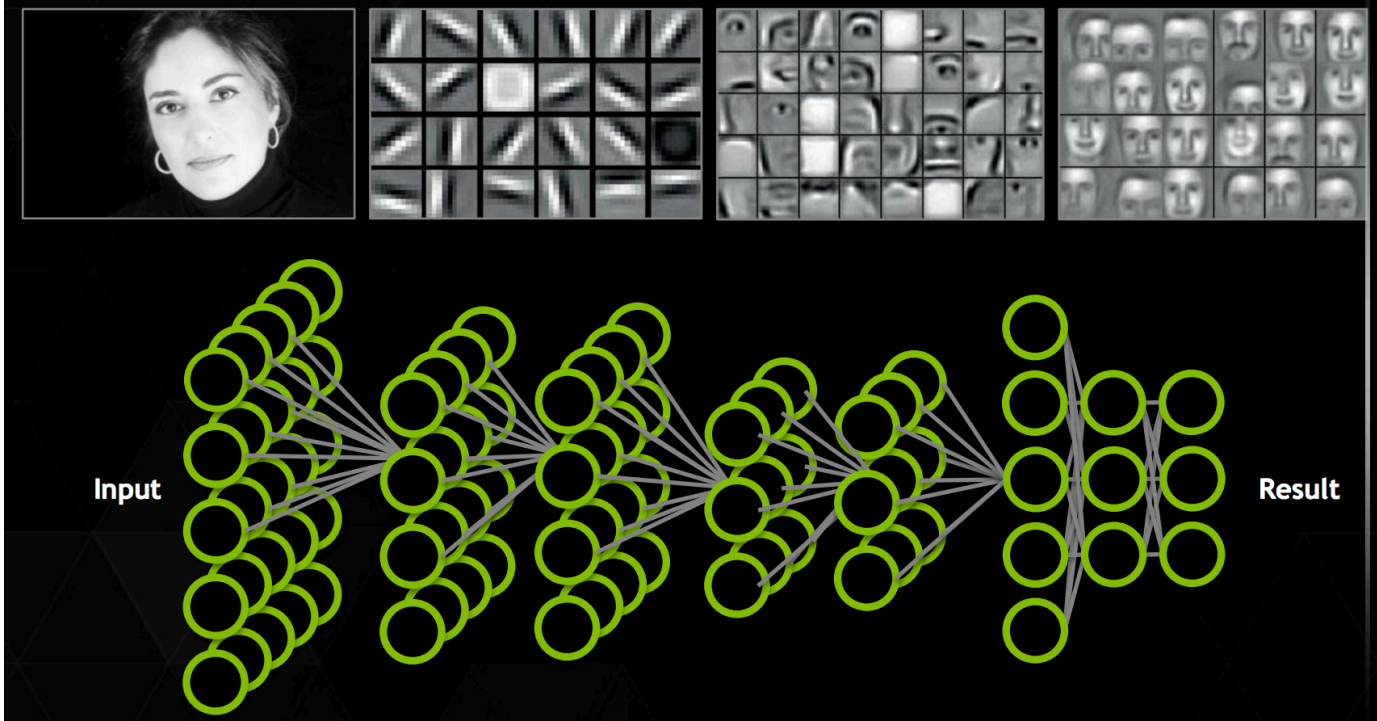
AI ? Machine Learning? Deep Learning?

Why is Facebook ideally positioned?

- Data collection at scale in online services
- Faster learning algorithms on new hardware, e.g. GPUs

HPC & Deep Learning

WHAT MAKES DEEP LEARNING DEEP?



Today's Largest Networks

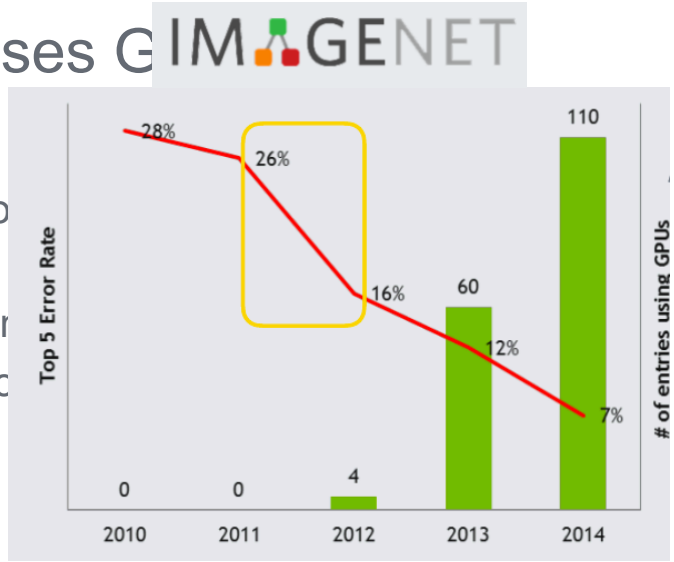
~10 layers
1B parameters
10M images
~30 Exaflops
~30 GPU days

Human brain has trillions of parameters - only 1,000 more.

HPC Implications of Deep Learning

Incredible pure acceleration^2 in the past 5 years

- Almost every entry in ILSVRC Imagenet uses G



- HW performance gains:
  - 2012: 2-GTX 580 (Alexnet), 2 x 1.6 = 3.2 Tflops/s single precisio
  - 2014: 4-Tesla K40, 4 x 4.3 = 17.2 Tflops/s single precision
  - 2015: 8-Maxwell (Bigsur) 8 x 5.6 = 44.8 TFlops/s single precisio
  - 2016: NVIDIA's Pascal, DGX-1 = 170 TF/s half precision in a bo
  - 2017(?): Volta …
- Algorithmic and Implementation gains
  - 2012: unfolded matrix multiplication (torch im2col)
  - 2014: cuDNN direct convolutions [Chetlur2014], fbfft convolutions [Vasilache2014]
  - 2015: tiled FFT implementations, Winograd convolutions [Lavin2015]
  - 2016: low precision Winograd convolutions running close to peak (cuDNN, Nervana)

HPC Implications of Deep Learning

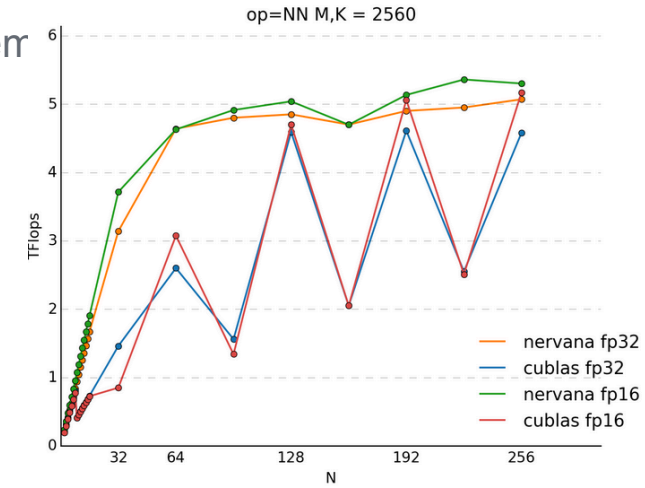
Faster Pace of Innovation in DNN Techniques and HW

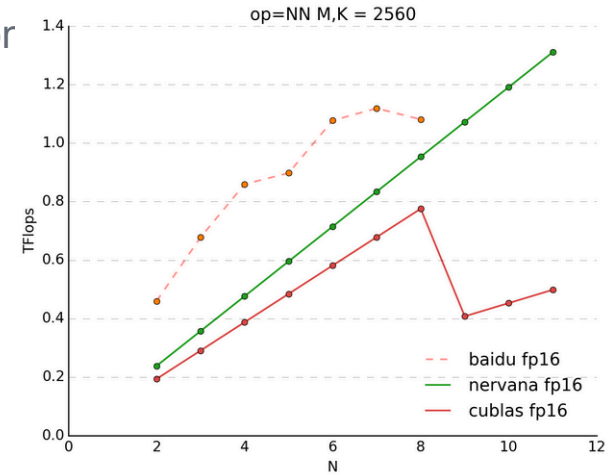
- Algorithmic innovations
  - Deep compression, trained networks fit in cache[Han2015]
  - Distributed Deep Learning [Zhang2014]
- Tooling innovations
  - Nervana's MaxAS assembler allows writing at the SASS level directly (register banks, operand reuse in ALU)
- Hardware innovations
  - DaDianNao: A Machine-Learning Supercomputer [Temam2014]
  - Nervana ISA [Nervana2015], acquired by Intel
  - Sparse, pruned connections and compression in HW [Han2016]

HPC Implications of Deep Learning

Pushing GPU hardware to the limit

- RNN example:
  - Sequence of small, dependent, matrix-matrix operations
  - Ideally sequence of small, dependent, matrix-vector operations (when batch size N=1), latency-bound regime
  - Implem ls or nentation feat.





Distributed Deep Learning

Distributed Deep Learning at FB

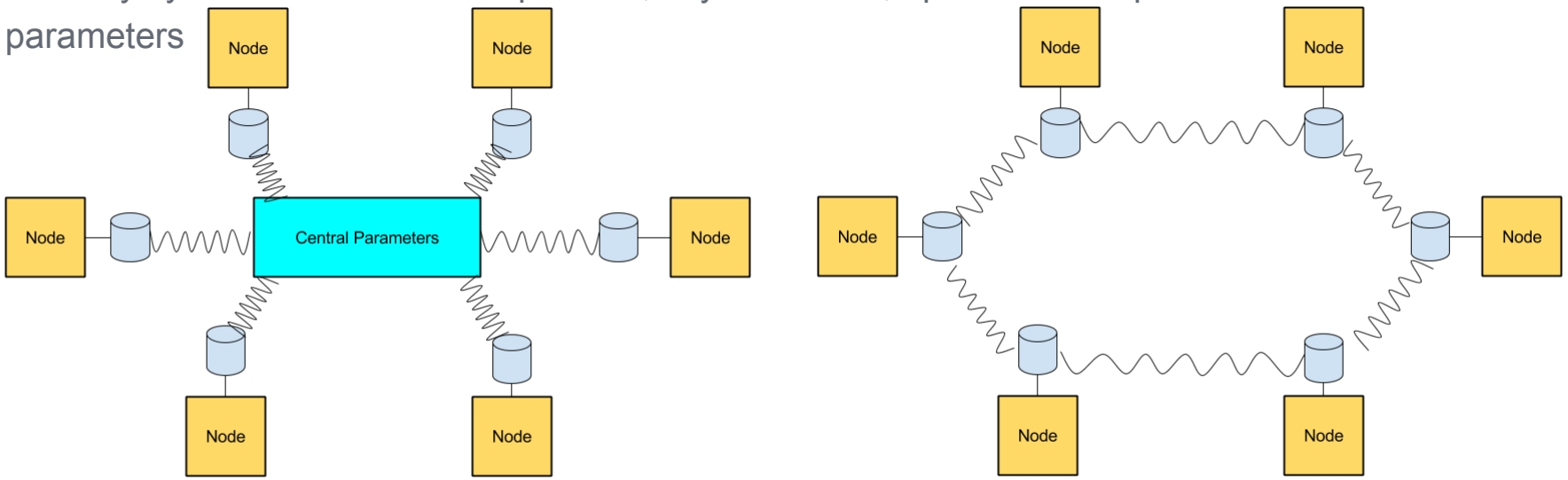
Disto

- Scala-like environment for Lua
  - Implemented on top of folly::Future (*https://github.com/facebook/folly*)
  - Multiple Lua threads per process, each with its own stack
  - Communication, serialization services via apache::Thrift (https://thrift.apache.org/)
  - Based on FAIR fblualib (*https://github.com/facebook/fblualib*)
  - Torch tensors are first class citizens and communicated specially (cuda IPC, etc) (http://*torch.ch*)
- Brings future/promise-style programming to Lua
  - Eventing mechanism depends on libevent (http://*libevent.org*)
- Also tied to FB internal RPC framework for Thrift
  - and other FB internal tools and infrastructure
  - likely won't be open-sourced

Distributed Deep Learning at FB

Elastic Averaging SGD

- New algorithm [Zhang2014]
  - Communication and coordination of work based on elastic force
  - Distributed parameters linked to center parameters and oscillate in its neighborhood
  - Locally synchronous SGD with periodic, asynchronous, updates Node parameters <-> Central parameters



Distributed Deep Learning at FB

EA-SGD

- Empirical speedup
  - Roughly follows sqrt(N), where N is the number of nodes up, measured up to 32 nodes
- Setup comprises:
  - 32 nodes, 4 Tesla K40 GPU per node
  - Data-parallel within node, EA-SGD across nodes
  - Sharded parameters distributed across 64 CPU-only servers
- No noticeable communication cost with published settings

[Zhang2014]

- Thanks to asynchronous prefetching and delayed updates (tau = 10 iterations, prefetch = 5 iterations)
- AlexNet trained with 5 full ”distributed” epochs (i.e. each nodes sees 5 full epochs in randomized order)

Very Large-scale Similarity Search

FB data and embeddings

Post embedding

Video embedding

Text embedding (word2vec)

$x \in \mathbb{R}^d$

Face embedding

User embedding

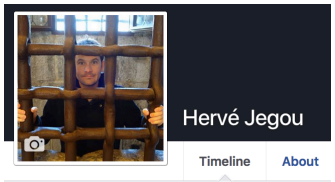


Image Embedding (CNN layer)

Compressed domain Similarity Search

Billions of vectors per query

Three (contradictory) performance criteria

- search quality
- speed
- memory usage

Bottlene

k for

Emb

$y_1, y_2, \ldots, y_n \in \mathbb{R}^d$

Indexing

Index in RAM

$x \in \mathbb{R}^d$

Result:

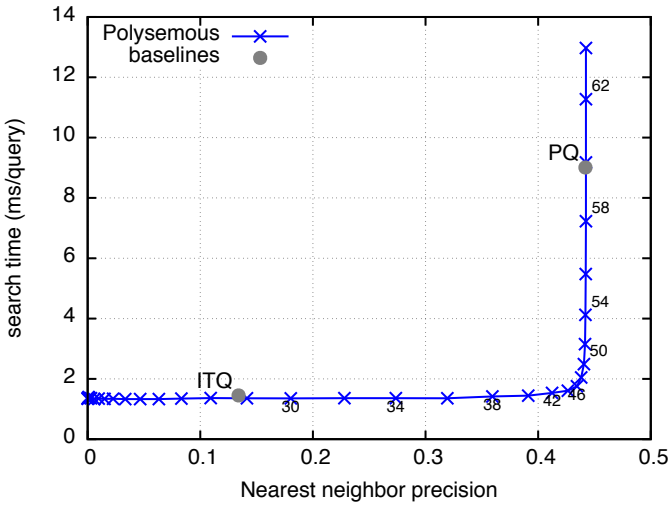
$i = \text{argmin}_{i=1..n} \|x - y_i\|^2$

Compressed domain Similarity Search

Our research outperforms all published results on **billion**-sized academics benchmarks (SIFT1B and Deep1B)

Deep1B (20 bytes/vector)			
method	hardware	1-R@1	time (μs/query)
Babenko @CVPR'16	1 thread	*0.450*	*20000*
Facebook @ECCV'16	1 thread	0.456	**3660**

SIFT1B (8 bytes/vector)			
method	hardware	1-R@10	time (μs/query)
Wieschollek @CVPR'16	Titan X	*0.350*	*150*
Facebook @ECCV'16	1 thread	0.349	485
	20 threads	0.349	35



Open Research

How FAIR works

We do research in the open:

- Publish research in best conferences and journals and make them available to the whole research community

- Contribute to open source initiatives such as Torch

➔ makes our research better and faster

➔ creates community around tools

➔ See https://research.facebook.com/ai