Interactive Machine Learning

- and our new INRIA project

John Canny

Computer Science Division

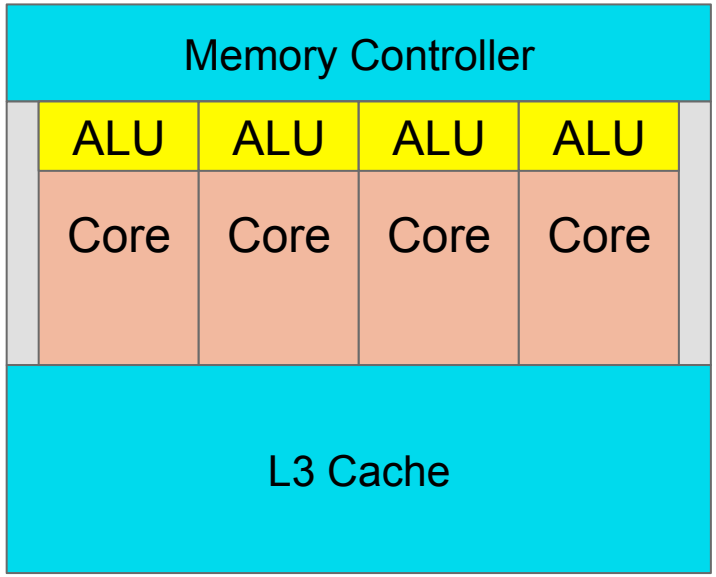
University of California, Berkeley

June, 2014

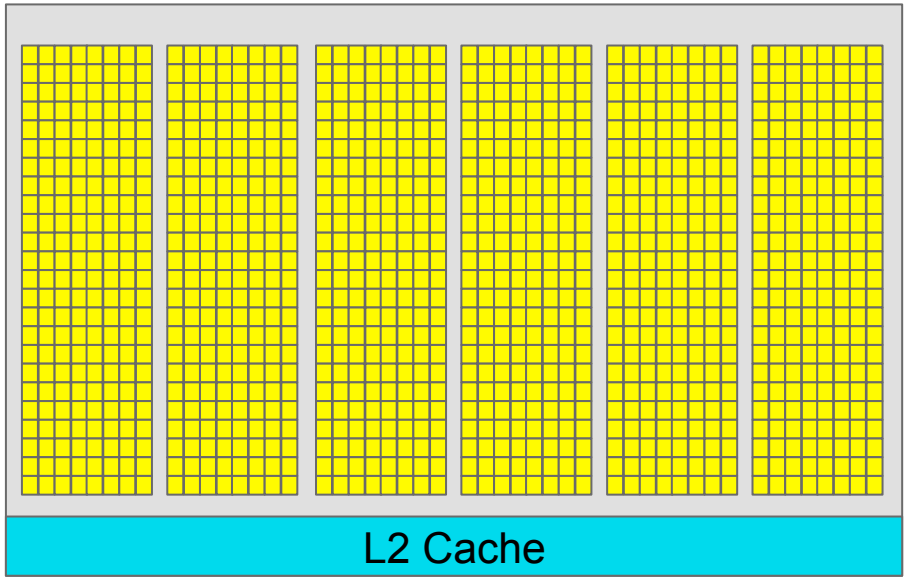
Where is my computer ?

Where is my computer ?

Intel® CPU

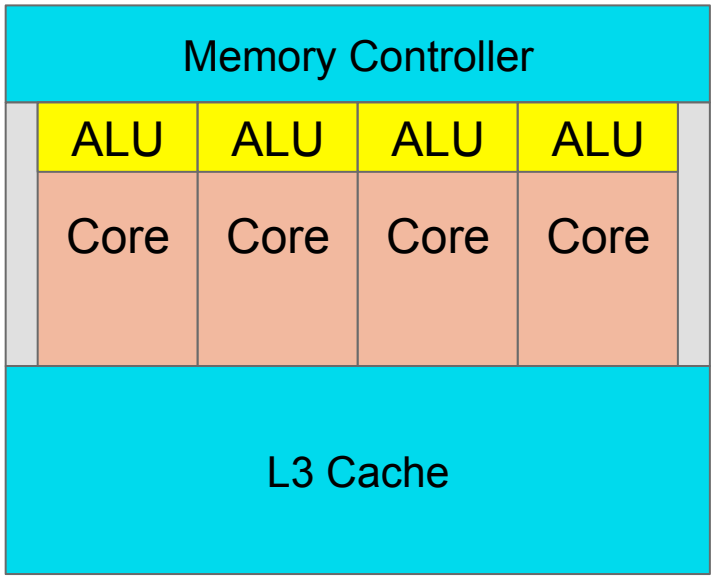


NVIDIA® GPU

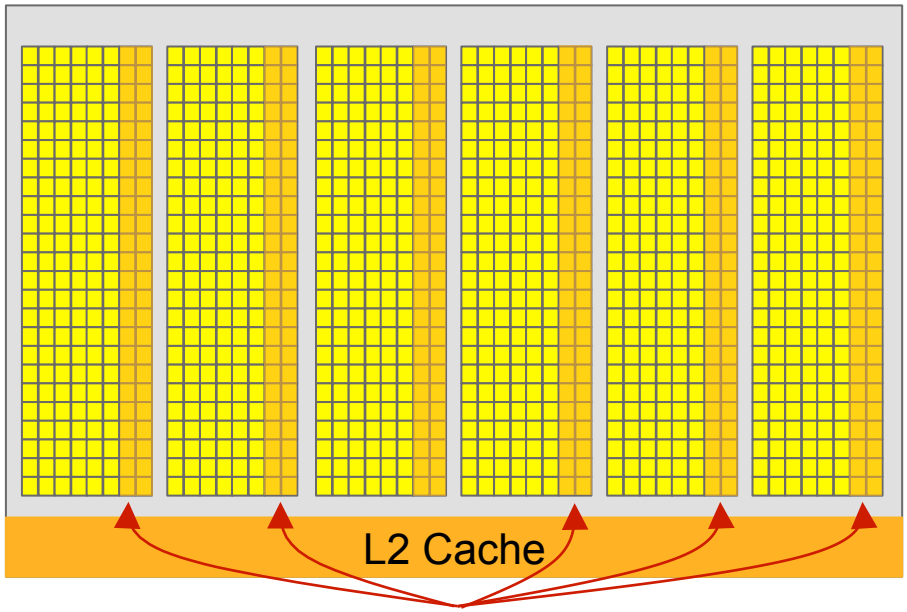


Where is my memory ?

Intel® CPU



NVIDIA® GPU



Hardware transcendentals (power series)

4 MB register file

4kB registers:

DDR3 memory (20 GB/s)

GDDR5 memory (200 GB/s)

A Quiz

Which is Faster:

A On a machine X, we:

Insert 10 million items (key,values) in a hash table

Lookup another 10 million items in the table

B Send items and queries over network to Machine Y

Move from Y's main memory to a GPU

Sort on GPU to align queries and keys

Resort to put queries back in original order

Move back to main memory from GPU

Send results back over network to Machine X

A Quiz

Which is Faster:

A On a machine X, we: **3s**

Insert 10 million items (key,values) in a hash table

Lookup another 10 million items in the table

B Send items and queries over network to Machine Y

Move from Y's main memory to a GPU

Sort on GPU to align queries and keys

Resort to put queries back in original order **1.8s**

Move back to main memory from GPU

Send results back over network to Machine X

A Quiz

Which is Faster:

A On a machine X, we: **3s**

Insert 10 million items (key,values) in a hash table

Lookup another 10 million items in the table

B Send items and queries over network to Machine Y **0.8s**

Move from Y's main memory to a GPU **0.04s**

Sort on GPU to align queries and keys **0.05s**

Resort to put queries back in original order **0.05s**

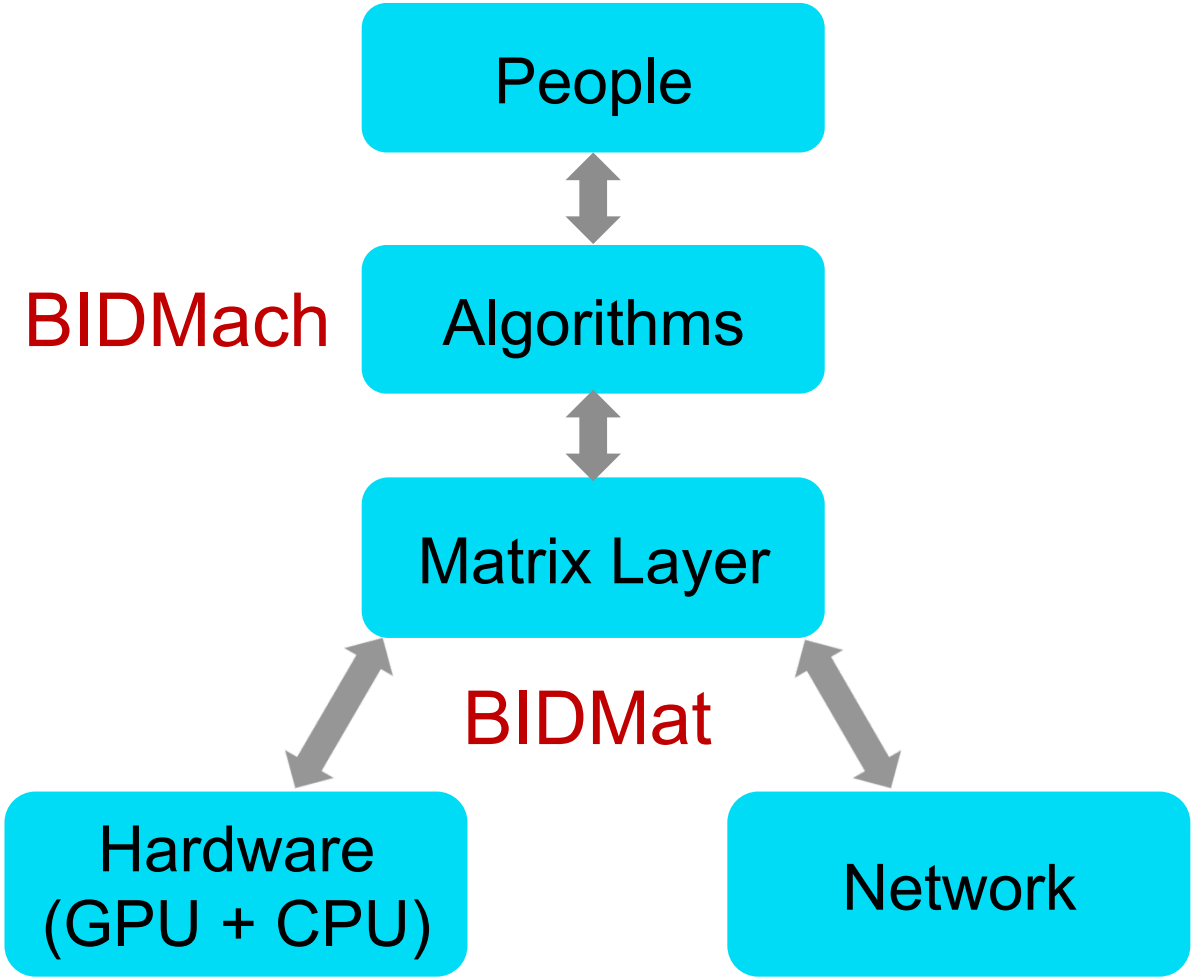
Move back to main memory from GPU **0.04s**

Send results back over network to Machine X **0.8s**

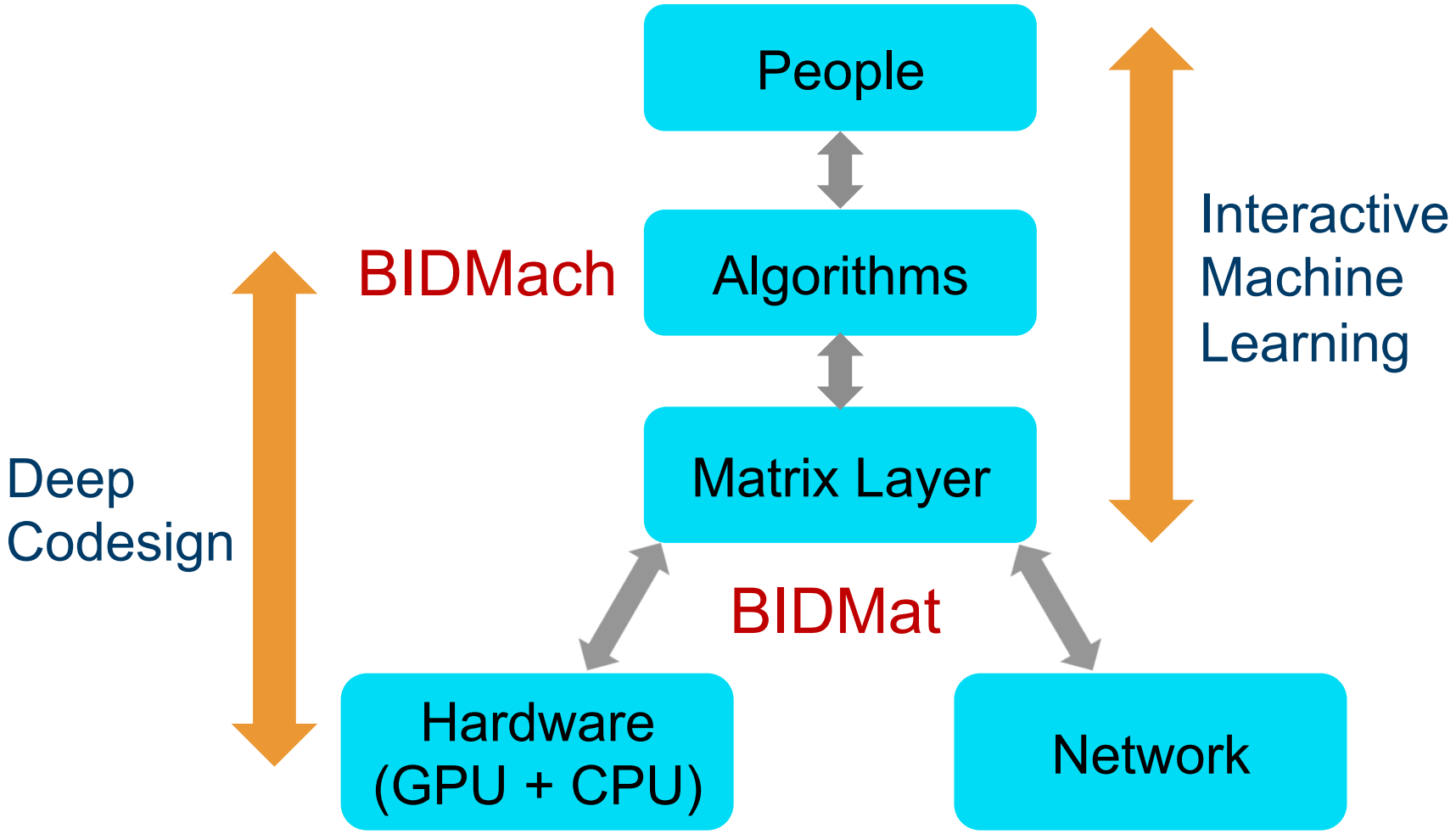
Take-aways

- **Location, Location, Location:** It really matters where data are and how they are accessed.
  - Hash tables (random access) very slow, lists, trees possibly slow.
  - Batched operations potential orders of magnitude faster.
- **Network Communication** is fast for large blocks of data.
- **Graphics processors** can do more than math (sorting, joins, grouping). “the GPU is the new CPU”
- Constants matter.

A response: BIDMach and BIDMat

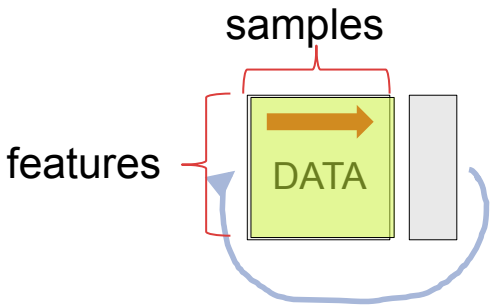


BIDMach and BIDMat



Trends in Machine Learning

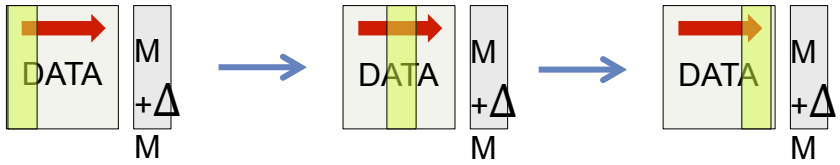
Classical: Batch model update in memory



Hadoop/Mahout
Spark: UC Berkeley
HaLoop: U. Washington

- **Incremental-update Methods**
  - Stochastic Gradient Descent (SGD)
  - Gibbs Sampling (GS)

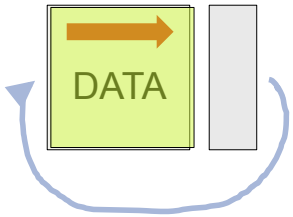
Large Datasets: Mini-batch model updates



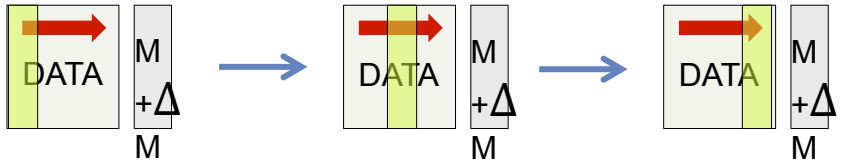
BIDMat/BIDMach: (this talk)
Downpour SGD: (Google)
Hogwild: U. Wisc.-Madison
Vowpal-Wabbit: Yahoo/MS

Trends in Machine Learning

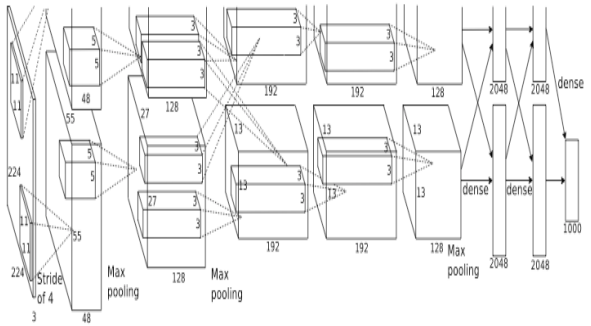
Classical: Batch model updates



Large Datasets: Mini-batch model updates



Deep Learning



Convnet, RNNLib, Visual-RBM: Toronto
Torch7: (NYU, NEC)
Theano: Montreal
Caffe: UC Berkeley

A GPU-enabled Matrix Toolkit

Written in the beautiful Scala language:

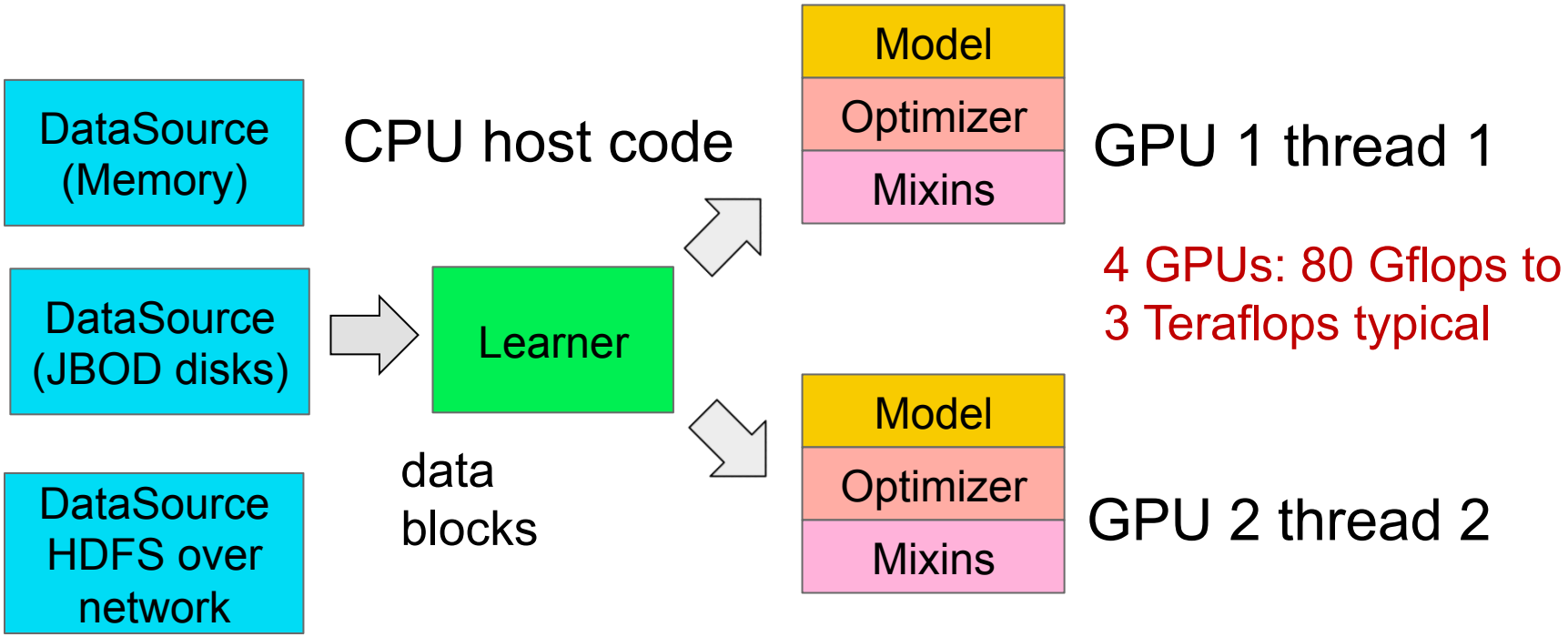
- Functional support, and lambda expressions.
- Interpreter, w/ excellent performance.
- Natural syntax +,-,*, °,•,⊗ etc and high-level expressivity.
- CPU and GPU backend (generics).
- Hardware acceleration – **many custom GPU kernels**
- Easy threading (Actors).
- Java VM + Java codebase – runs on Hadoop, Spark.
- Good text processing, integrated XML interpreter.

Inspired by Matlab, R, SciPy

A modular learning Toolkit

BIDMACH

Zhao+Canny
SIAM DM 13, KDD 13, BIGLearn 13



Compressed disk streaming at
~ 1.5GB/s ≅ 40-100 Hadoop nodes

:
:

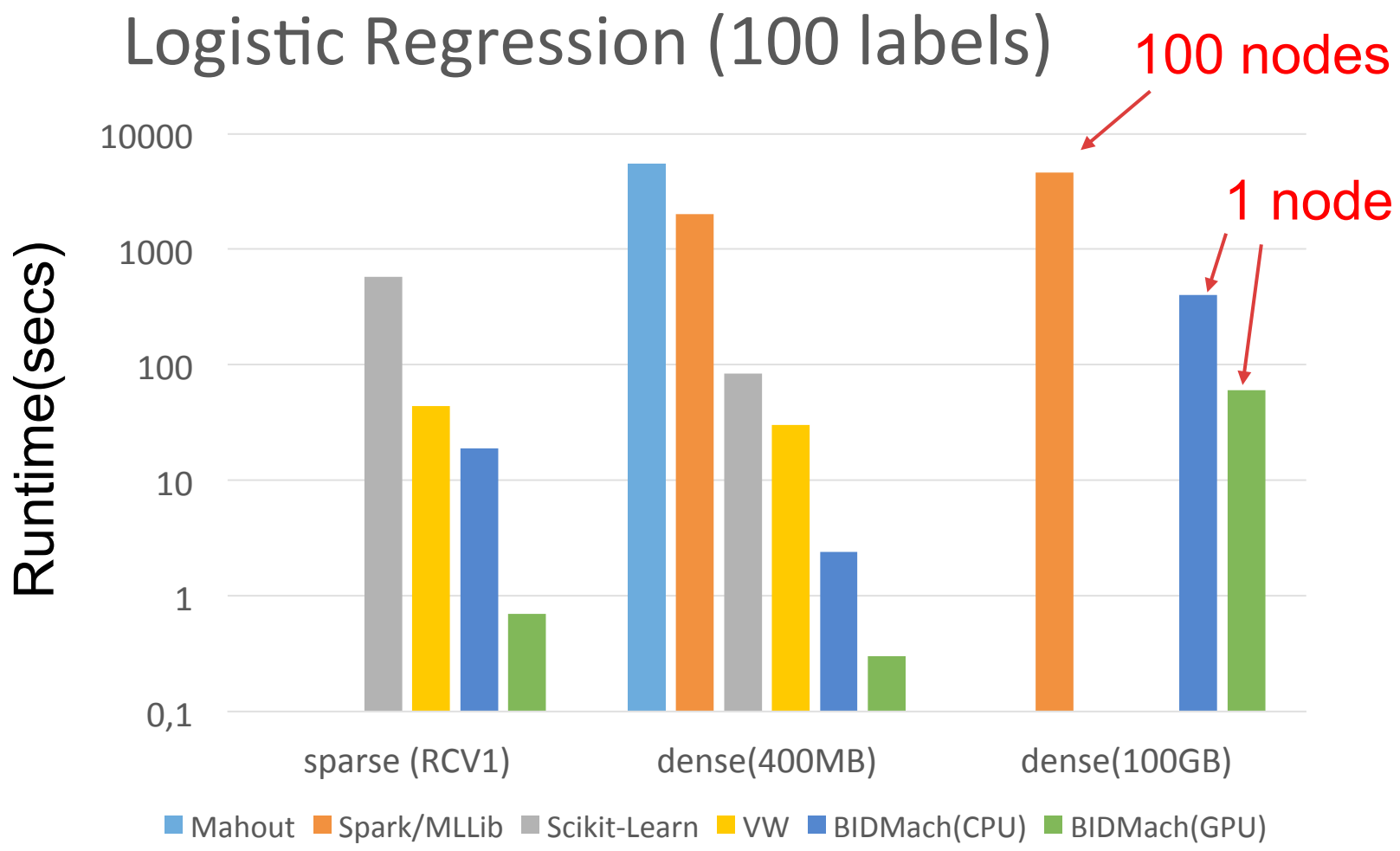
BIDMach Approach

- Full GPU integration, generic matrix classes:
  - Dense-Sparse
  - Single-Double precision
  - CPU or GPU implementation
- Use locality:
  - Dense and Sparse matrices, instance-major ordering
  - Packed, traversal-optimized trees
  - Partitioning and blocking of sparse matrices and tensors
- Minibatch updates:
  - Scalable, memory-efficient
  - Allows steady flow of information across the network

Top-10 “Big Data” Algorithms

1. Regression (logistic, linear) + Naïve Bayes
2. Support Vector Machines
3. Greedy Clustering (k-Means)
4. Topic Models (Latent Dirichlet Allocation)
5. Collaborative Filtering (Sparse Matrix Factorization)
6. Random Forests
7. Hidden-Markov Models
8. Spectral Clustering
9. Factorization Machines (Regression with Interactions)
10. Multi-layer neural networks
11. Natural Language Parsing

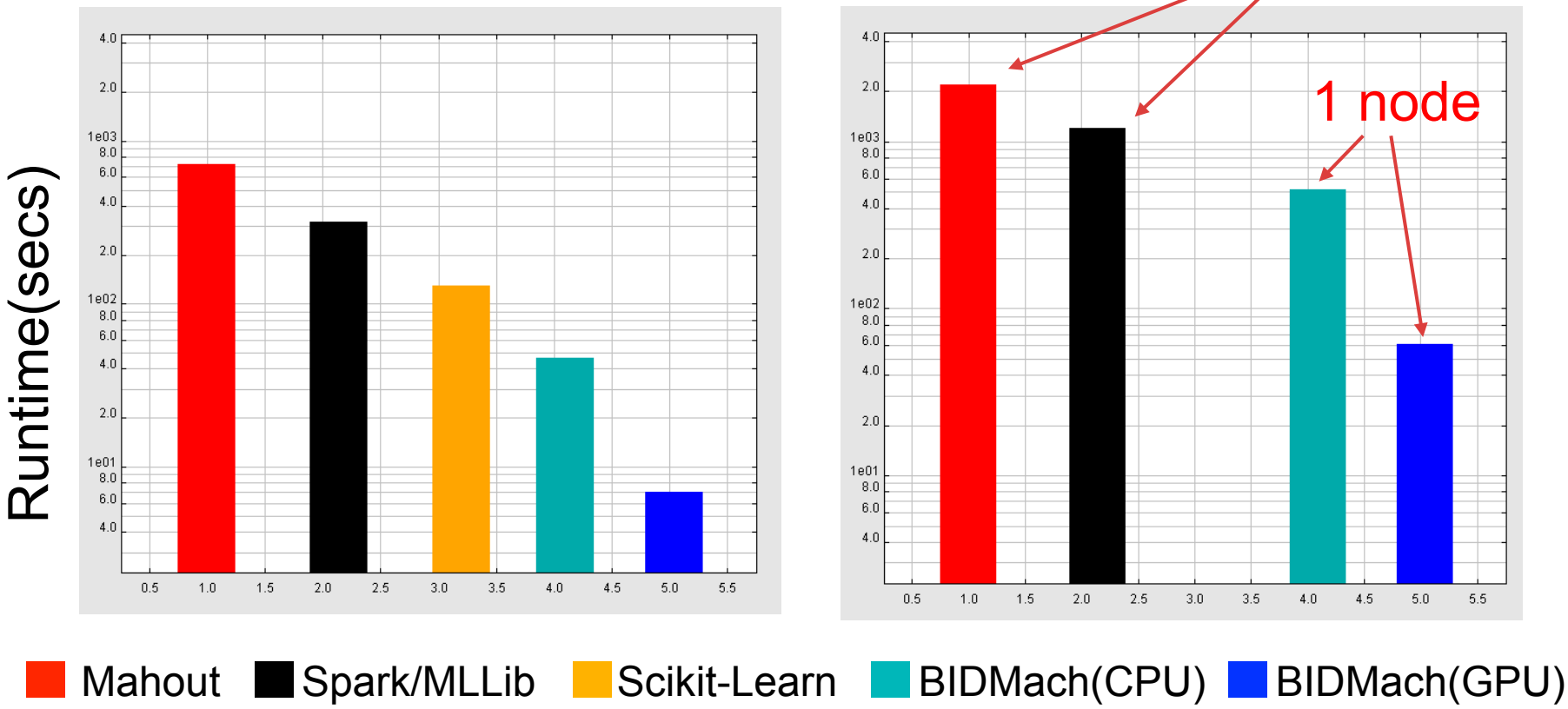
BIDMach Benchmarks



BIDMach Benchmarks

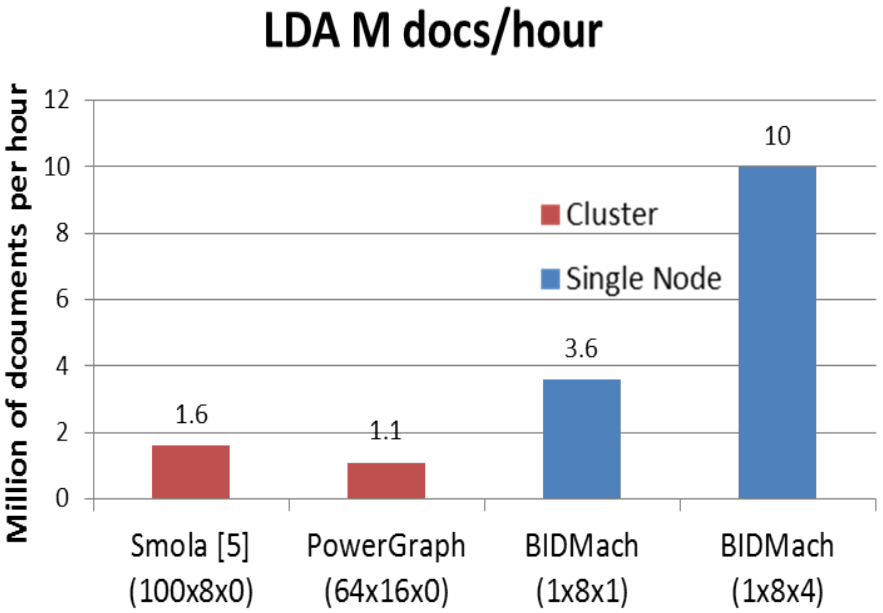
K-Means, single node
(400MB dense data)

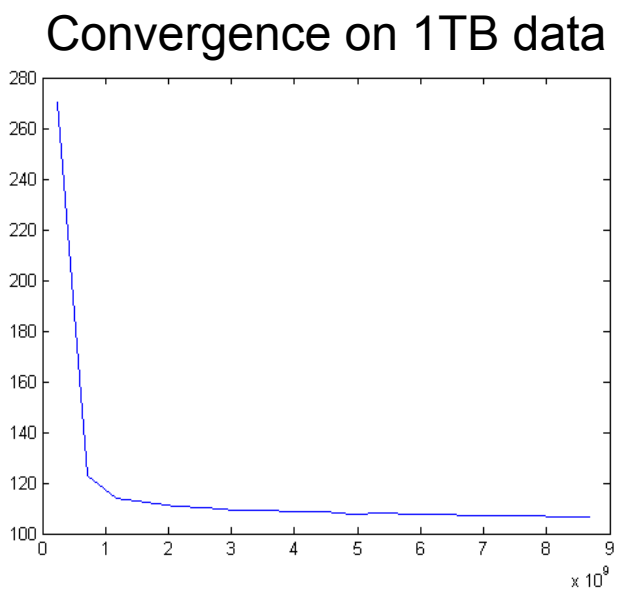
K-Means
(5GB dense data)



BIDMach Benchmarks

Variational Latent Dirichlet Allocation





BIDMach outperforms cluster systems on this problem, and has run the largest topic models to date (up to 10 TB), on one node.

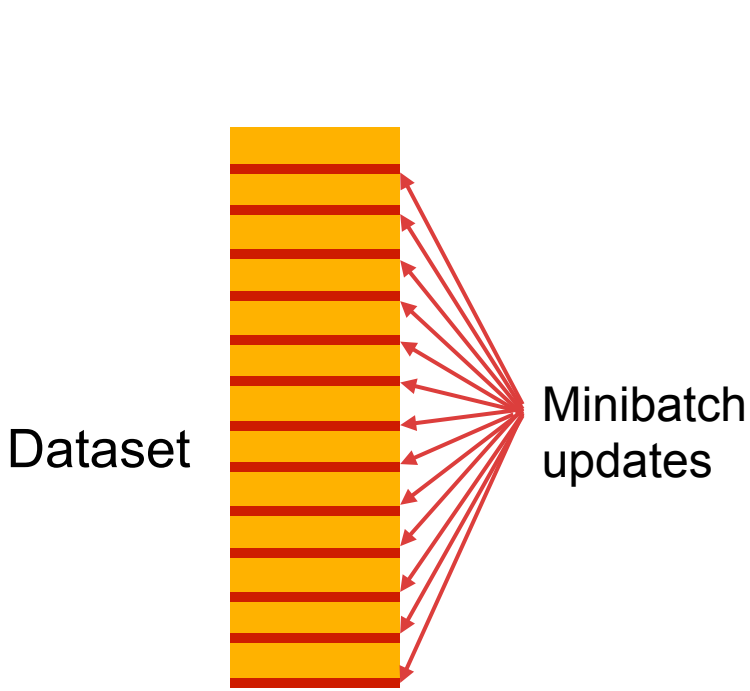
Power Budget

- BIDMach with GPU is faster than other single-machine systems by 2-3 orders of magnitude.
- BIDMach also outperforms cluster systems with 40-100 nodes by 0-2 orders of magnitude.
- But cluster power consumption grows with the number of nodes, and in terms of Mflops/W, BIDMach is 2-4 orders of magnitude more efficient.

Scaling up

Can we combine node acceleration with clustering?

Its harder than it seems:

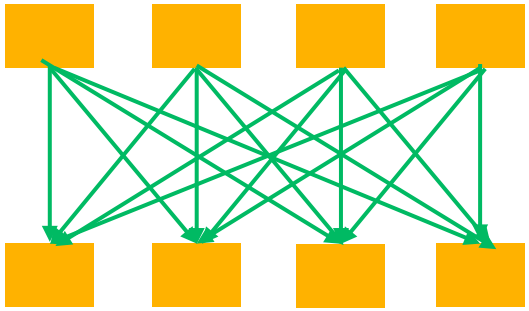


Single-node Minibatch Learner

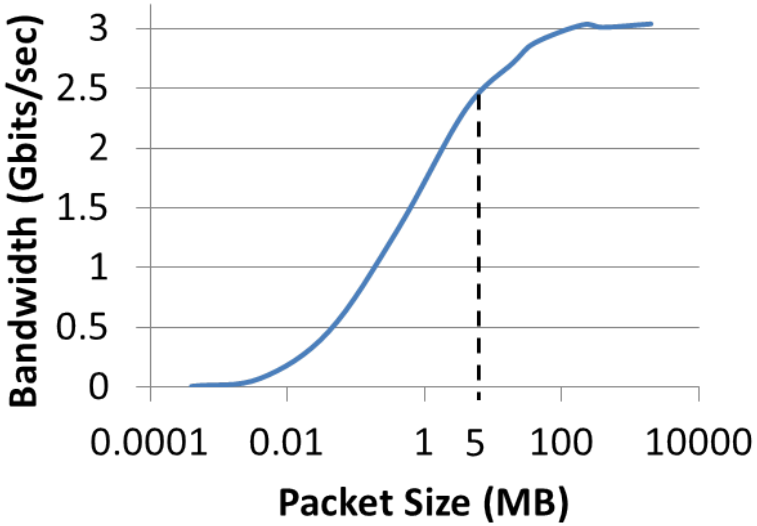
Cluster-based Learner

What about MapReduce?

Most MapReduce implementations (Hadoop, Spark, Powergraph*) **don't scale**. i.e. The communication time stops decreasing and starts increasing past a certain point, reversing the advantages of parallelism.



During reduce, N nodes exchange $O(N^2)$ messages of diminishing size.



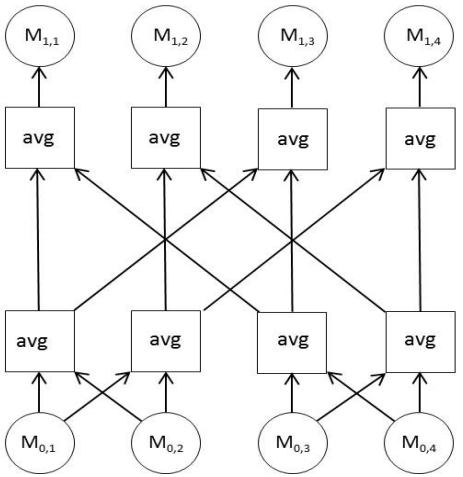
Butterfly Mixing

Zhao and Canny. In SIAM Data Mining, 2013.

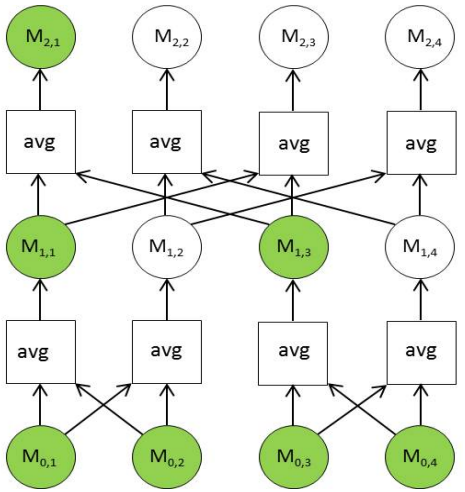
Many minibatch algorithms are lag-tolerant.

Butterfly mixing interleaves computation and communication.

Butterfly AllReduce



Butterfly Mixing



Circles: model update

Squares: aggregates

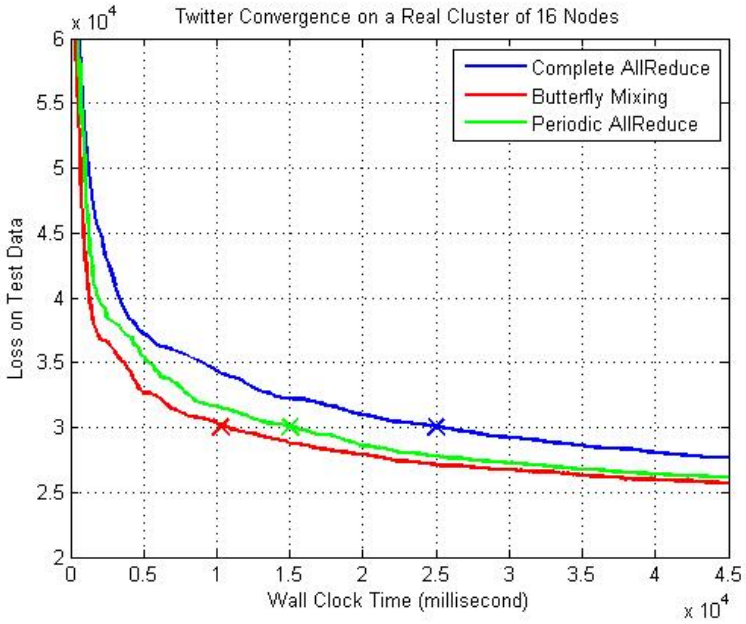
Diagonal edges: network messages.

Butterfly Mixing: SVM convergence

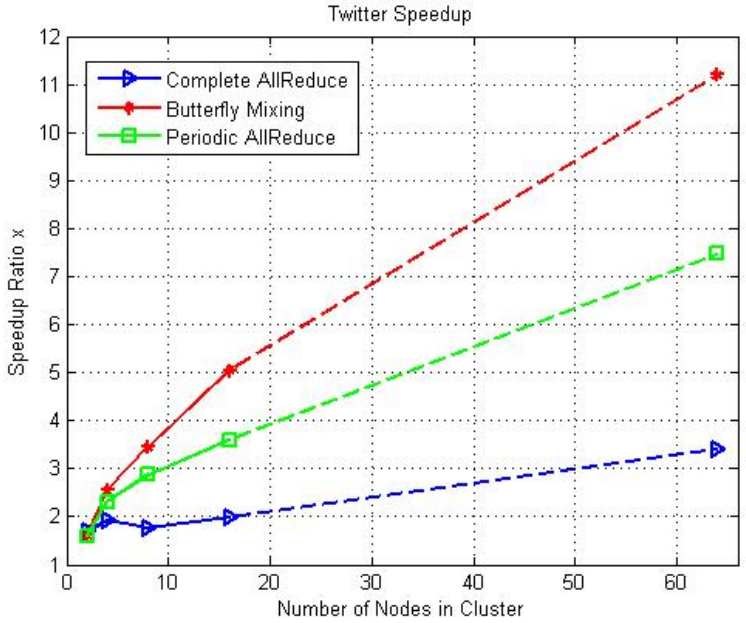
Convergence almost as fast as **full allreduce** every step.

- Tweets sentiment classifier with 250k uni-gram features
- 170 million unique tweets labeled with emoticon.

Convergence

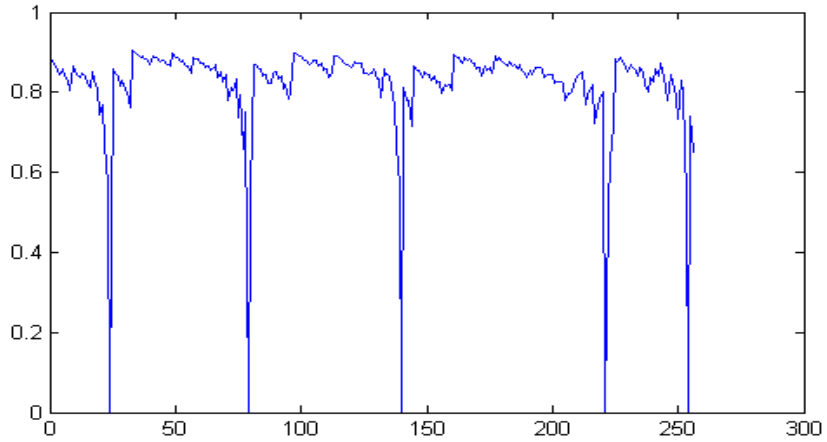


Scaling Behavior



Butterfly Mixing fault tolerance

Graph of weights of data from other nodes, on a 256-node network with 5 failures:



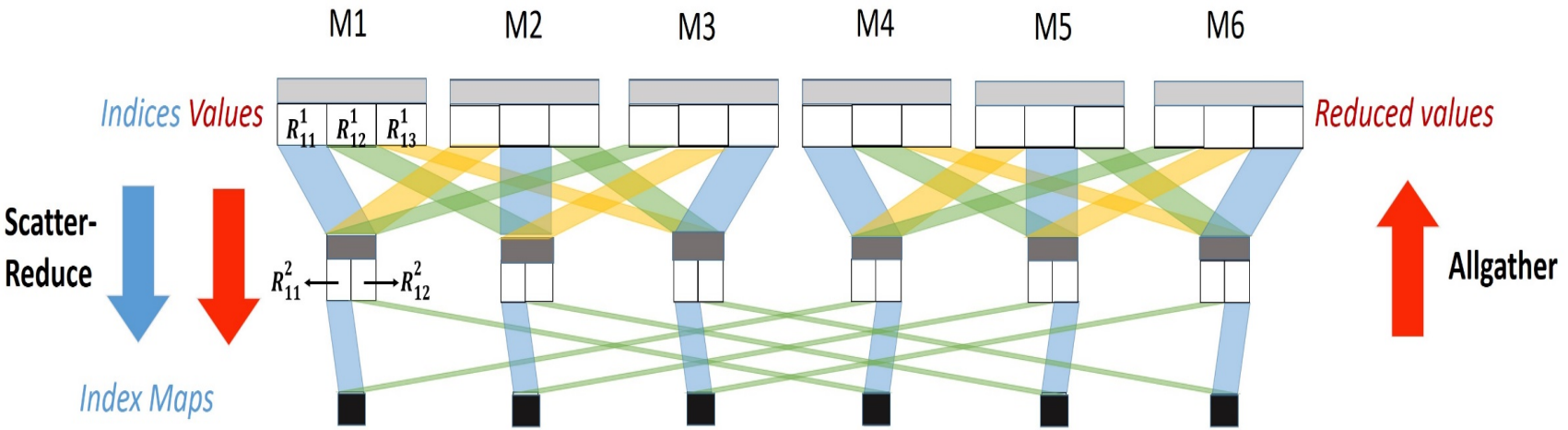
Efficiency (i.e. size of equivalent averaging network / size of live network) is 99%.

What about Graph Data?

- Behavioral graphs (e.g. social networks, web graphs etc) are not efficiently separable and challenging to parallelize.

Kylix: a Sparse Butterfly Allreduce

- Kylix uses a sparse, heterogeneous, nested butterfly network to perform reduces on graph data.
- Performance is near-optimal (within a factor of 2).
- It makes a large network “look” like a small one – top layer costs dominate.

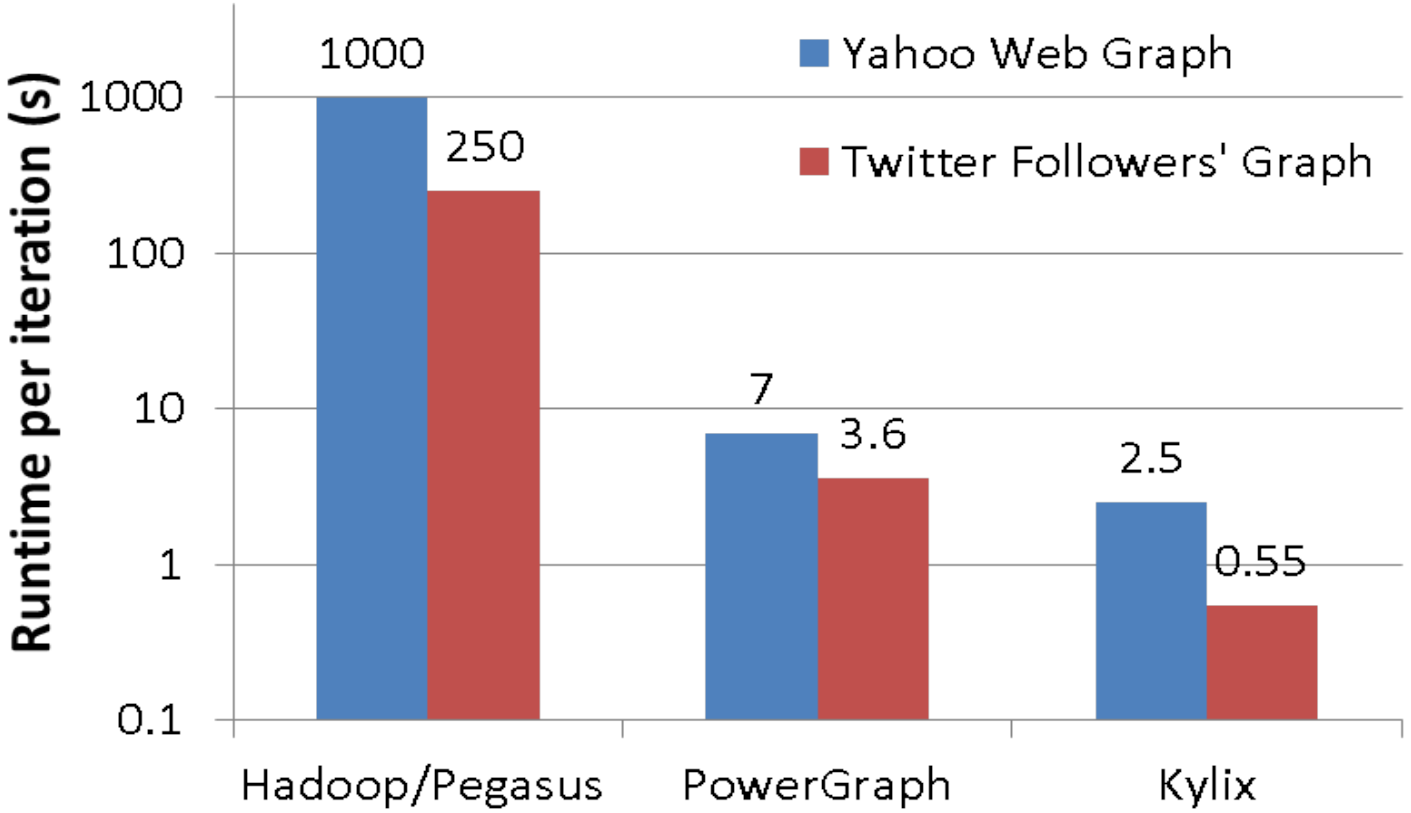


Zhao & Canny, ICPP 2014 (accepted)

Kylix: Benchmarks

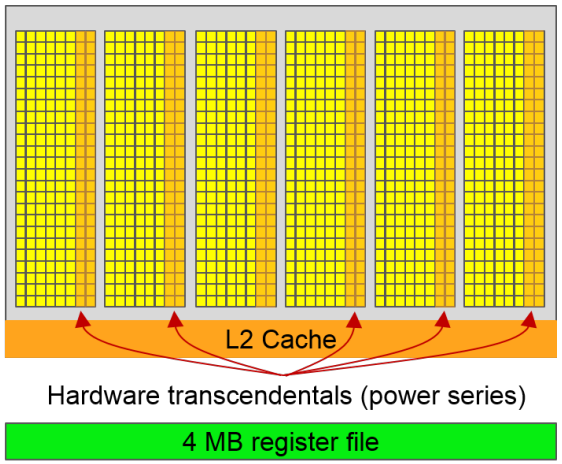
Computing Pagerank on large, power-law graphs.

All systems use 64 nodes



Current Work: Gibbs Sampling

- Gibbs sampling is a very general method for inference in machine learning, but typically slow.
- But it's a particularly good match for SIMT hardware (GPUs), we are seeing 100x speedups.

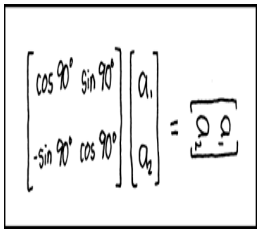


- We can further accelerate sampling using:
  - Parameter cooling: sampling as optimization.
  - Factored approximate distributions and importance sampling.
- Sampling can be competitive in performance with symbolic methods, but it is simpler and may produce better optima.

Data Analysis as Craft

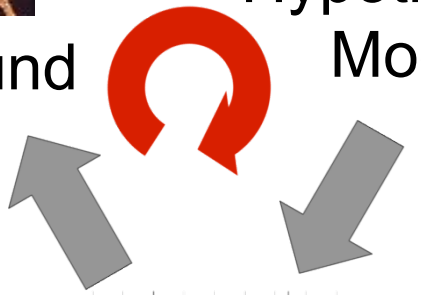
Analyst's Workflow

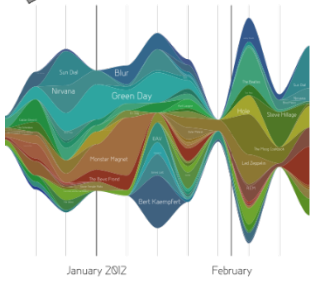
Digging Around in Data



Hypothesize Model

Large Scale Exploitation

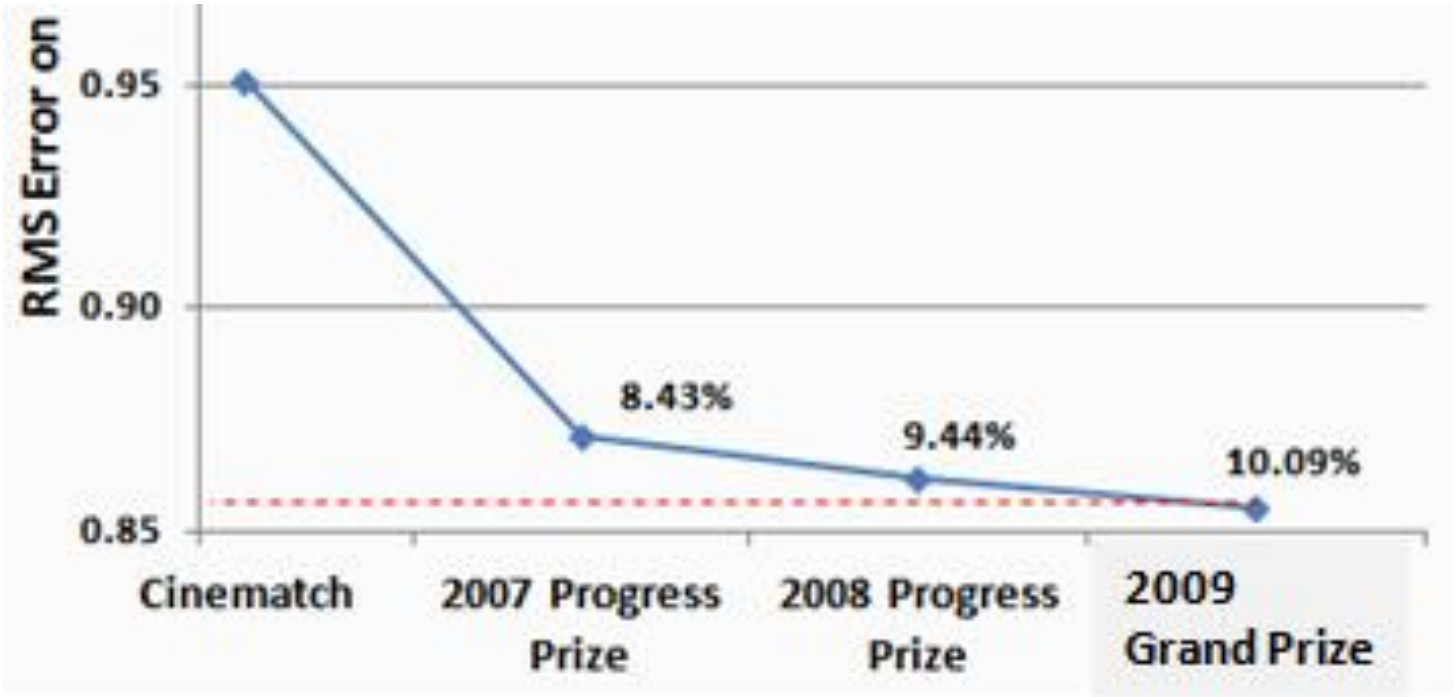




Evaluate Interpret

Iteration = Improvement

Netflix Prize Performance over time



Best entries were complex ensembles of many models

Interactive Machine Learning

- Turn iteration into interaction.
- Use bespoke likelihood functions to tailor the model (Mixins).
- Use stochastic search (Gibbs sampling), under human control, for “intelligent annealing.”
- Video

Software

Code:

github.com/BIDData/BIDMat

github.com/BIDData/BIDMach

Wiki:

http://bid2.berkeley.edu/bid-data-project/overview/

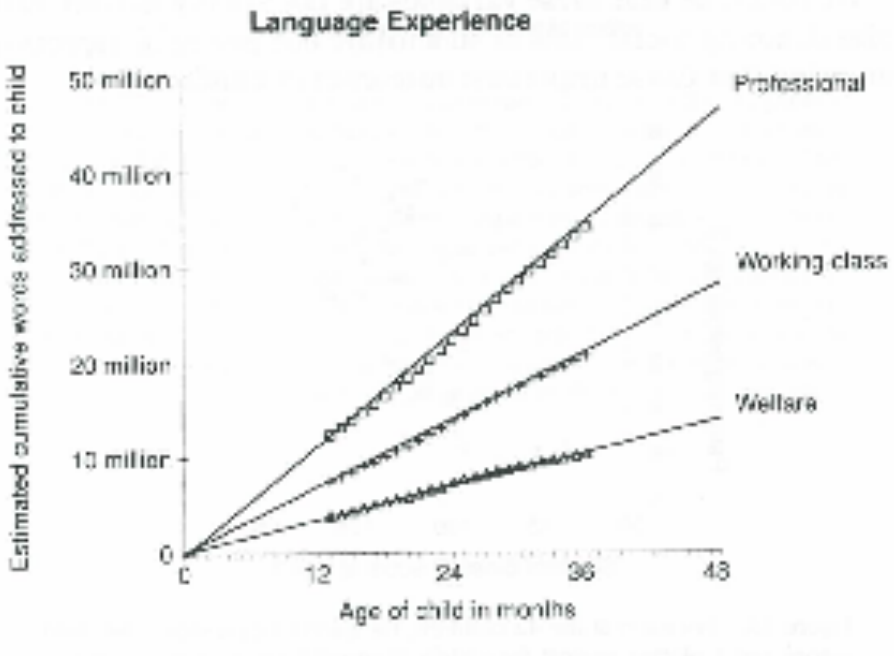
(includes binary bundles)

BSD-style open source libs and dependencies.

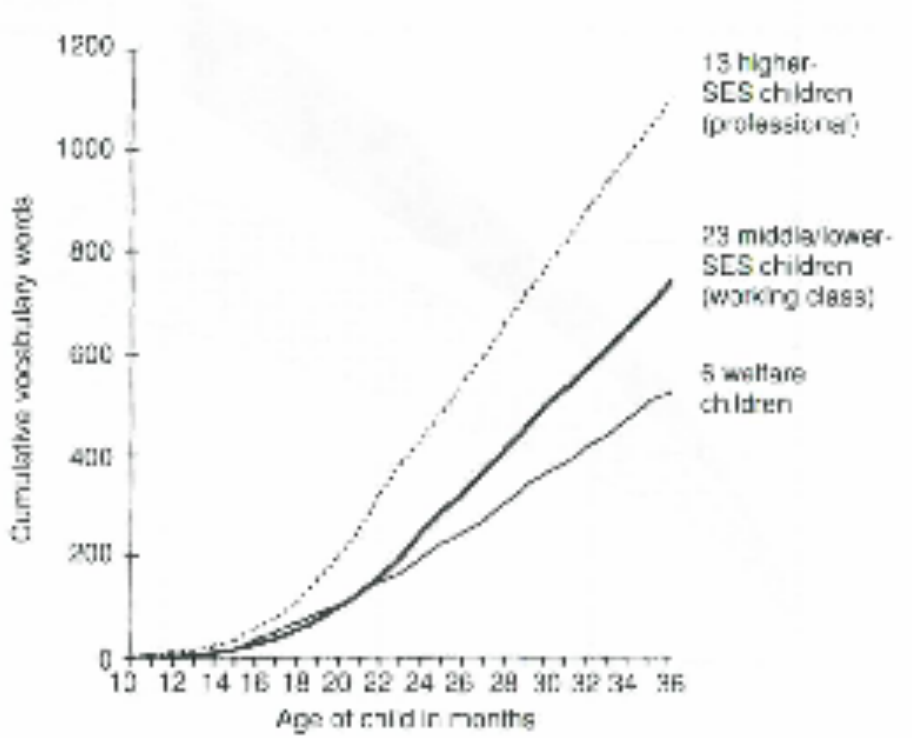
New Project: Language Learning

Language affects Learning

Language Exposure

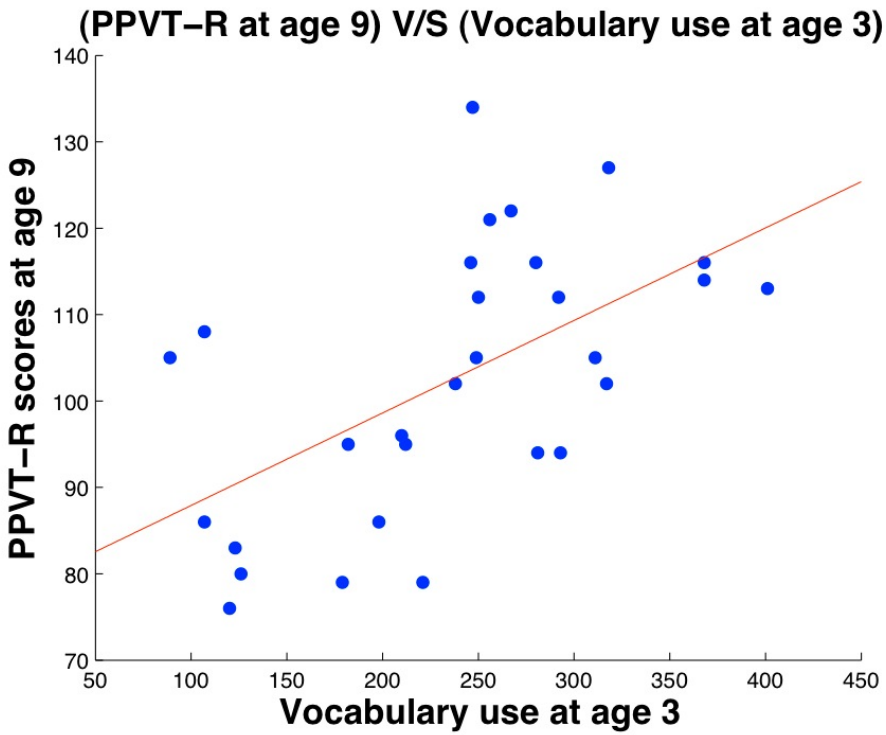


Language Use

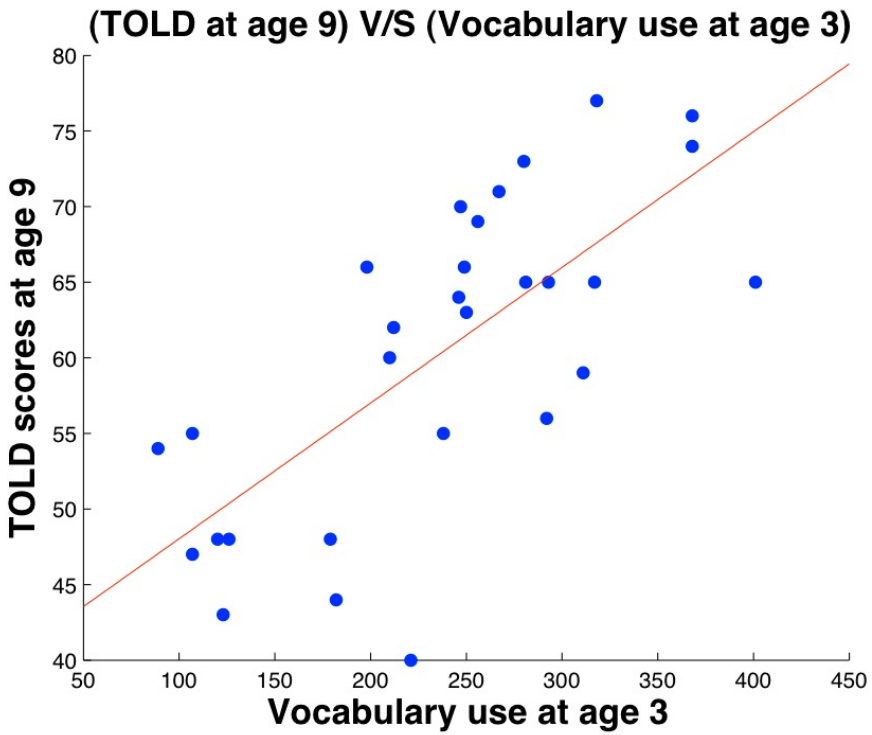


From Hart & Risely: ‘‘Meaningful Differences in the Everyday Experience of Young American Children”

Performance in middle school



r=0.62



r=0.74

Language Exposure

Virtually no books in low-SES households (less than 1 per household on average).

No practice of reading to children.

Limited language interaction between children and adults.

What can we do?

Children ask questions

- One recent study found 4-5 year olds asked 80 questions per hour.
- Question-asking is a natural part of cognitive development [Callanan 1992, Chouinard 2007]
- Questions provide experience with question construction, statements, explanation construction and complex tenses.

Spot: A Language Game

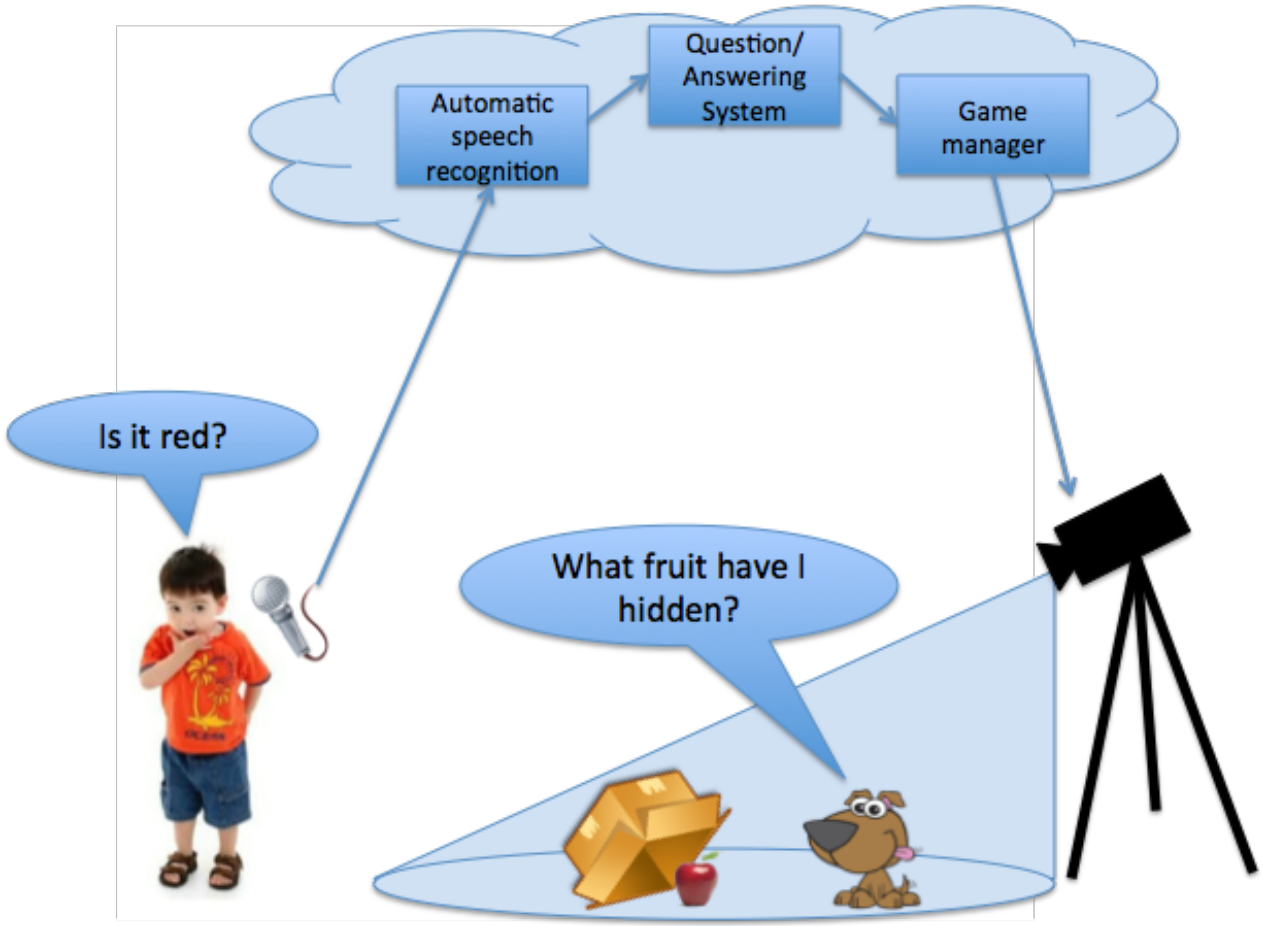
Tewari and Canny, CHI 2014
(Best Paper Honorable Mention)

- Uses a virtual character (a dog from the Sims pets) as game partner.
- Plays “20 questions”:
  - Spot shows the child two objects.
  - Spot hides one object, asks child to ask questions about it.
  - Child asks questions, spot responds.
  - Child tries to guess which object was hidden.

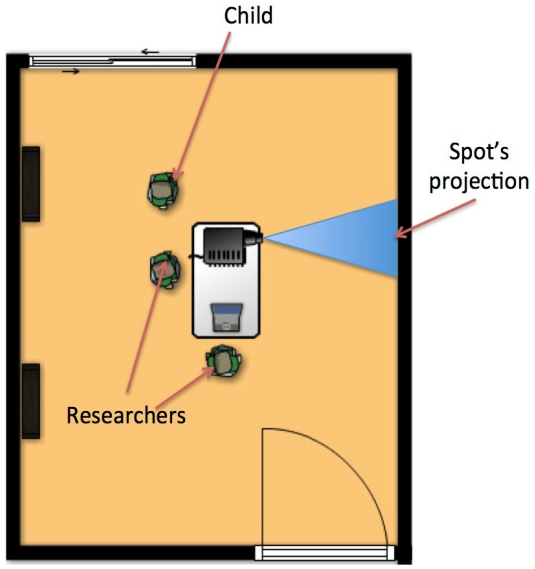
Spot: Goals

- Aims to develop:
  - Knowledge of language structure
  - Vocabulary – verbal and verbal/visual
  - Knowledge of object structure (parts), and attributes
- Designed for extended play:
  - Character is highly animated, in the child's world (projected on a wall)

Spot's "Mind"


Automatic speech recognition
Question/ Answering System
Game manager
Is it red?
What fruit have I hidden?

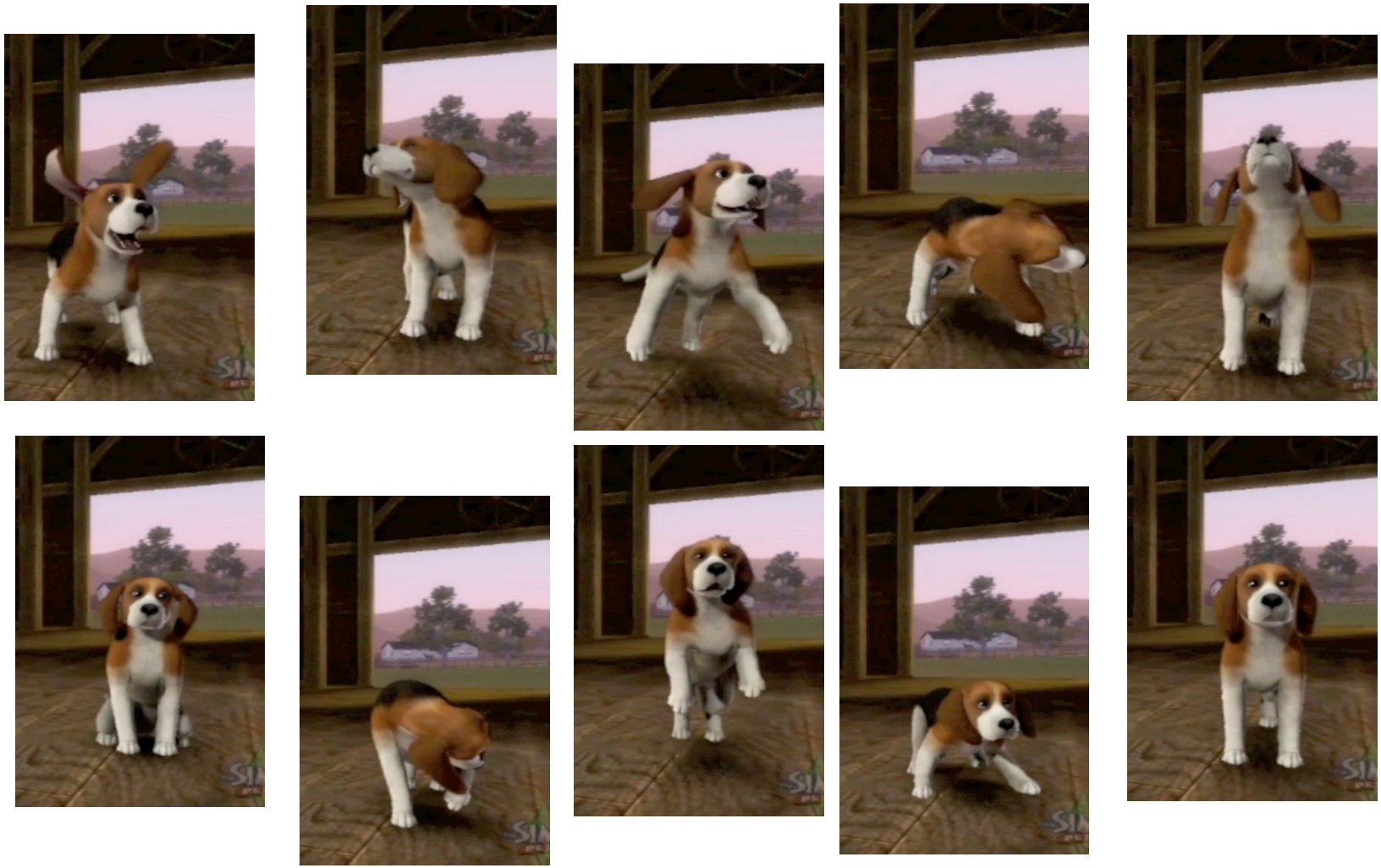
Spot goes to School



- Research Preschool, in a naturalistic experimental room.
- Participants - 10 male, 10 female (4-5 years)
- Typical session:
  - The child plays 20 questions with Spot projected on the wall.
- Data collection
  - Videos were transcribed and analyzed.

Animation Design

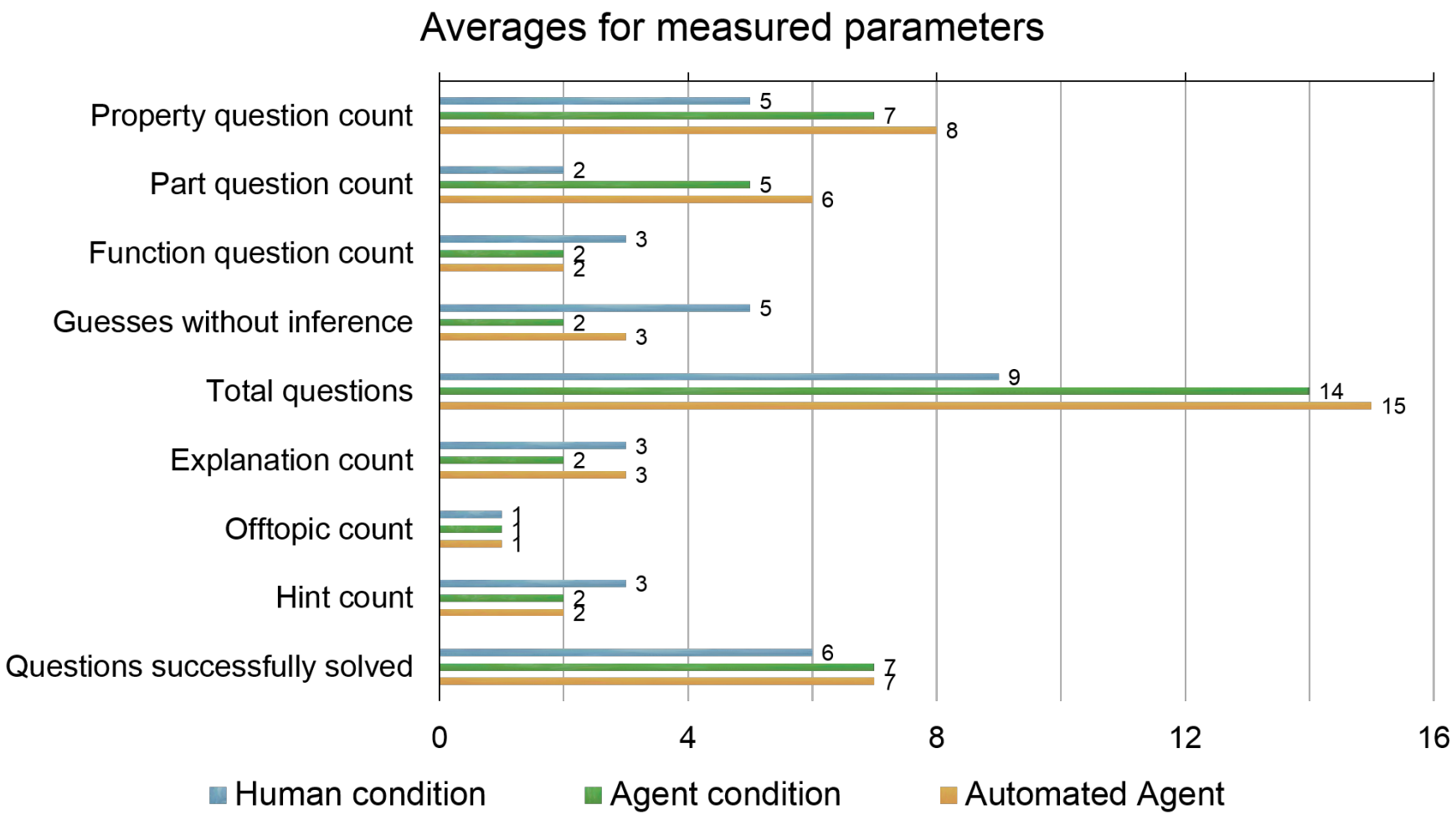
Uses “Machinima”: control of actions via the game + recording



Results

- Children ask questions, and a lot of them! (210 questions)
- The ones who asked more, solved more.
- Questions had categories (Part, Function, Property)
- Limited need for repeated explanation & limited tendency to guess without proper inference.
- Off-topic dialogues were limited (~5%).

Results vs. Human Partner



Speech Recognition is OK

Matching transcribed speech to question patterns:

- With perfect transcription - 6% (23/346)
- With Google speech recognition - 14% (49/346)
- Speech recognition errors are predictable - error lookup tables
- MER (error lookup) - 11%

Errors and Rules:

- “Does it meow?” - “Does it mean now?” - error lookup table.
- “Do you wear it on your head?” - handled by location attribute.
- “Does it have for?” - handled by edit distance.
- “A hat?” - Can’t handle (happens seldom), can be treated as a guess.

Going Beyond Spot – INRIA project

- Can we build an open-ended conversation partner that can talk about the child's world?

- Can we make it mobile, able to stay with the child, and perhaps move autonomously?

- Machine Vision:
  - Recognizing Objects
  - Recognizing Relationships
  - Recognizing Actions
  - Contextualizing (Activities)
- Natural language
  - "Read" and synthesize descriptions of scenes and video.

Going Beyond Spot – INRIA project

- Machine Vision and Natural Language have both seen rapid growth in “deep” neural processing algorithms.
- Its very natural to develop visual/textual vocabulary (as a child does) using integrated “reading” systems.

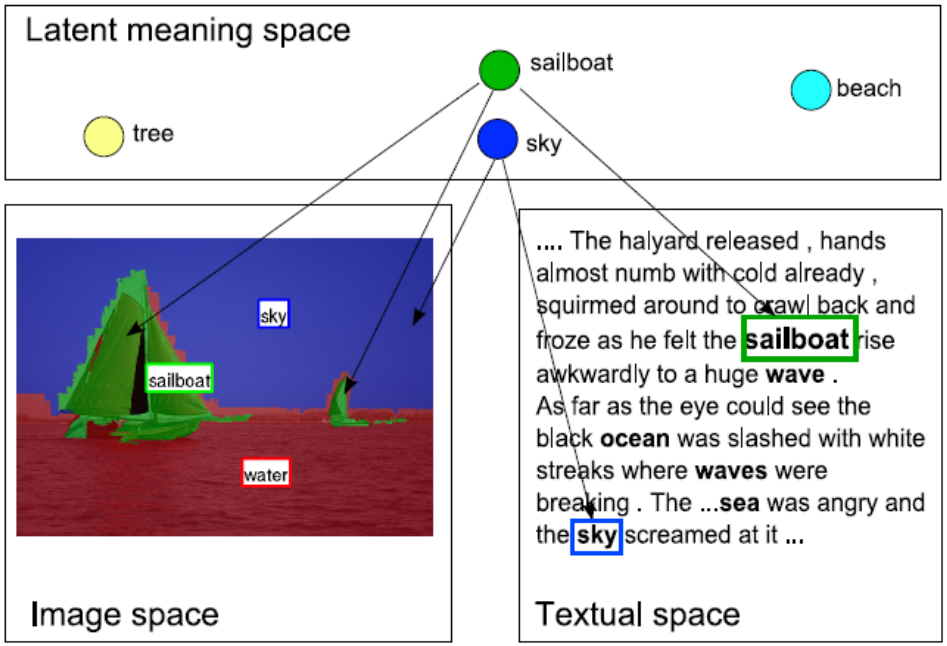
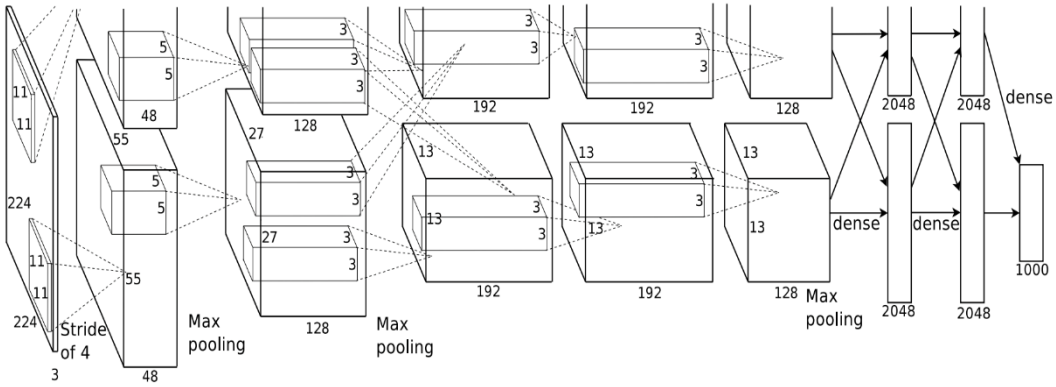


Figure from “Connecting Modalities: Semi-supervised Segmentation and Annotation of Images Using Unaligned Text Corpora. Socher and Li, CVPR 2010.

INRIA project - Approach

- Data-Driven Inference: Watson, Google speech recognition, Siri
- Machine Reading: DARPA, AI^2 solicitation
- Deep Representations:
  - Hold records for image recognition, speech processing
  - Layers represent progressively higher-level concepts
  - Can study linguistic and visual concepts together
  - A chance to study human and machine learning in parallel



The Toronto ImageNet network

INRIA project, Willow Group (Paris)

Year 1:

- Develop recognizers for inter-object relationships in images
- Develop relationship recognizers for text (e.g. rel-grams)

Years 2 & 3:

- Action recognition, including effects on relationships

Experimental Goal:

- “Read” i.e. produce semantic description of, children’s picture books.

Years 4 & 5:

- Work on dialog development for talking, asking and answering questions about the world around the child.

Experimental Goal:

- Build and test an embodied conversation partner (robot).