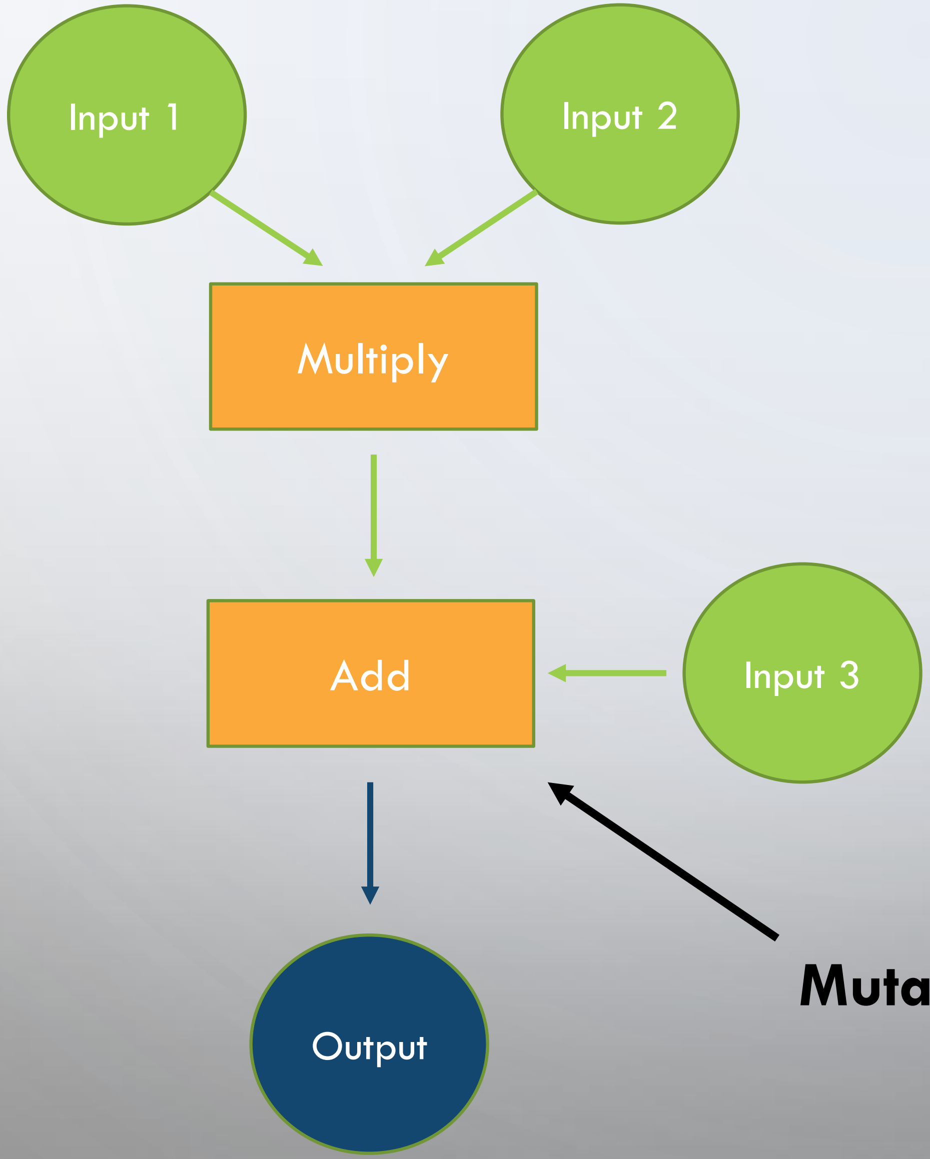
TENSORFLOW: A SYSTEM FOR LARGE-SCALE MACHINE LEARNING

AUTHORS: MARTÍN ABADI, PAUL BARHAM, JIANMIN CHEN, ZHIFENG CHEN, ANDY DAVIS, JEFFREY DEAN, MATTHIEU DEVIN, SANJAY GHEMAWAT, GEOFFREY IRVING, MICHAEL ISARD, MANJUNATH KUDLUR, JOSH LEVENBERG, RAJAT MONGA, SHERRY MOORE, DEREK G. MURRAY, BENOIT STEINER, PAUL TUCKER, VIJAY VASUDEVAN, PETE WARDEN, MARTIN WICKE, YUAN YU, AND XIAOQIANG ZHENG

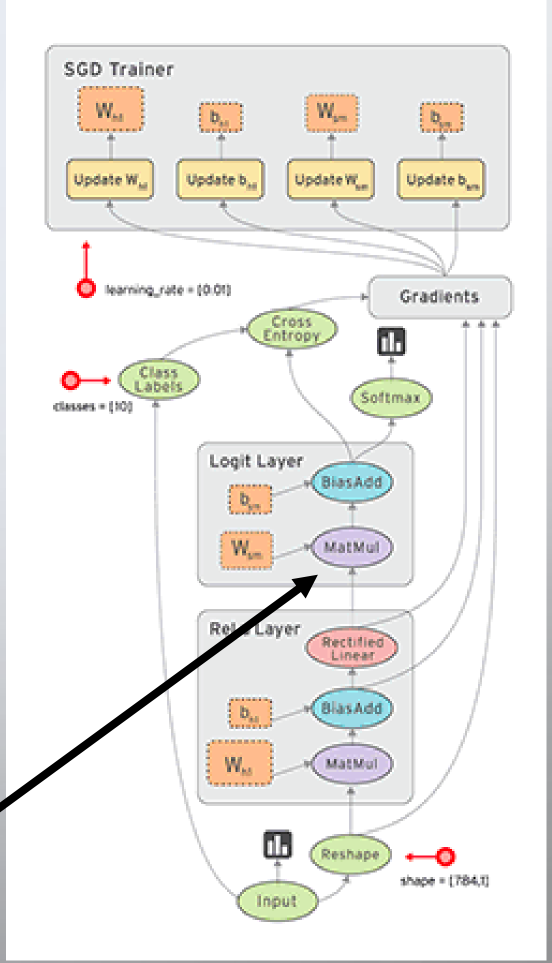
OVERVIEW

- Large Scale ML System
- Distributed Compute and Training
  - Multi-node
  - Heterogenous Environemnts
- Dataflow Graphs
- Open Source
- Mathematically Flexible
  - Bespoke Loss & Kernels
- Fault Tolerant

DATAFLOW GRAPHS

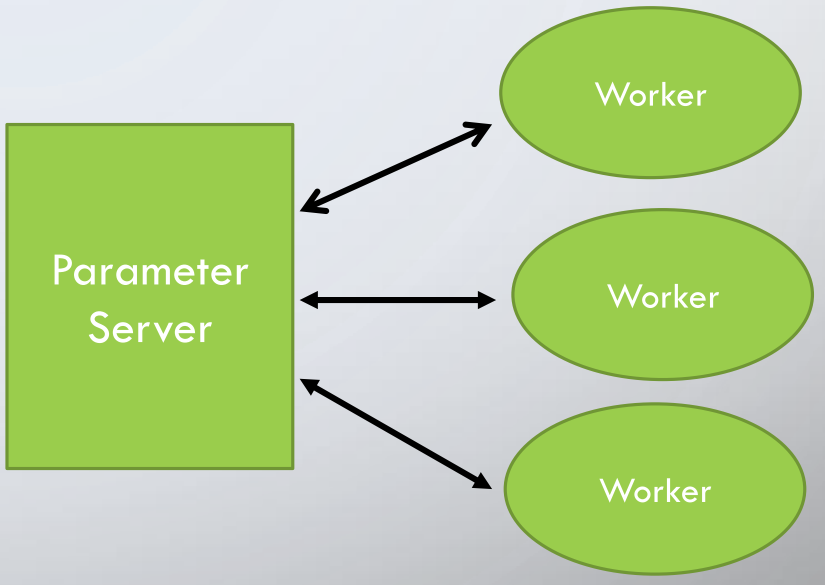


Mutability!

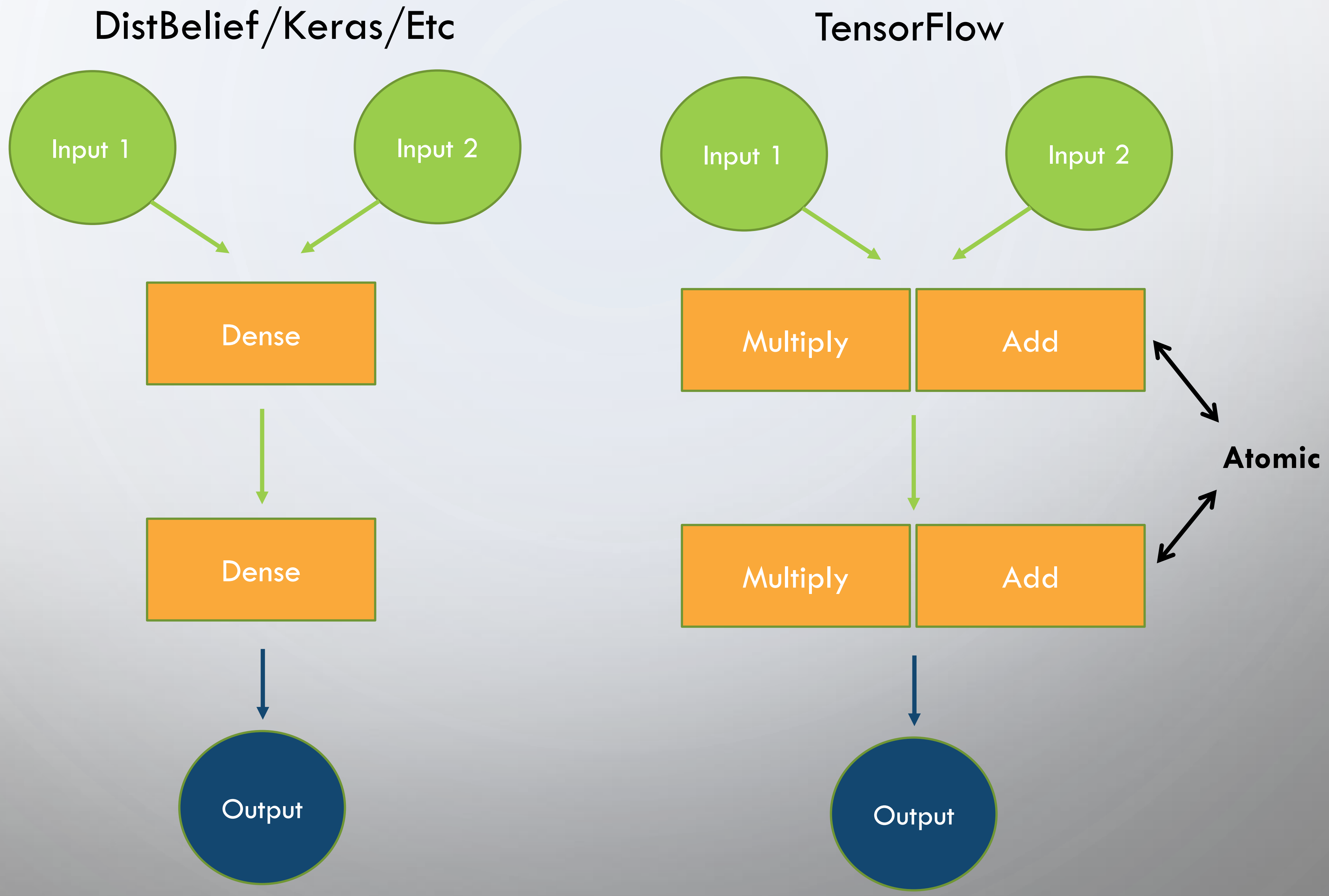


PRIOR WORK

- DistBelief
  - Architecture
    - Parameter Server
    - Workers
  - Inflexible Layers
  - Inflexible Training Algorithms
    - RNNs, LSTMs, GCNs challenging
  - Optimized for large clusters
- Caffe & Theano
  - Similar



TensorFlow is designed to improve flexibility!


DistBelief/Keras/Etc
Input 1
Input 2
Dense
Dense
Output
TensorFlow
Input 1
Input 2
Multiply
Add
Multiply
Add
Atomic
Output

ACCELERATOR ABSTRACTION

CPU

GPU

TPU

UNITS OF TENSORFLOW

- Graph
  - Subgraph → Partitioned subgraphs are distributed to individual compute devices
    - Edges
      - Tensors → Multidimensional arrays
    - Vertices
      - Operations → Add, Multiply, Sigmoid
- Automatic Partitioning
  - Subgraphs distributions maximize compute efficiency

CONTROL FLOW

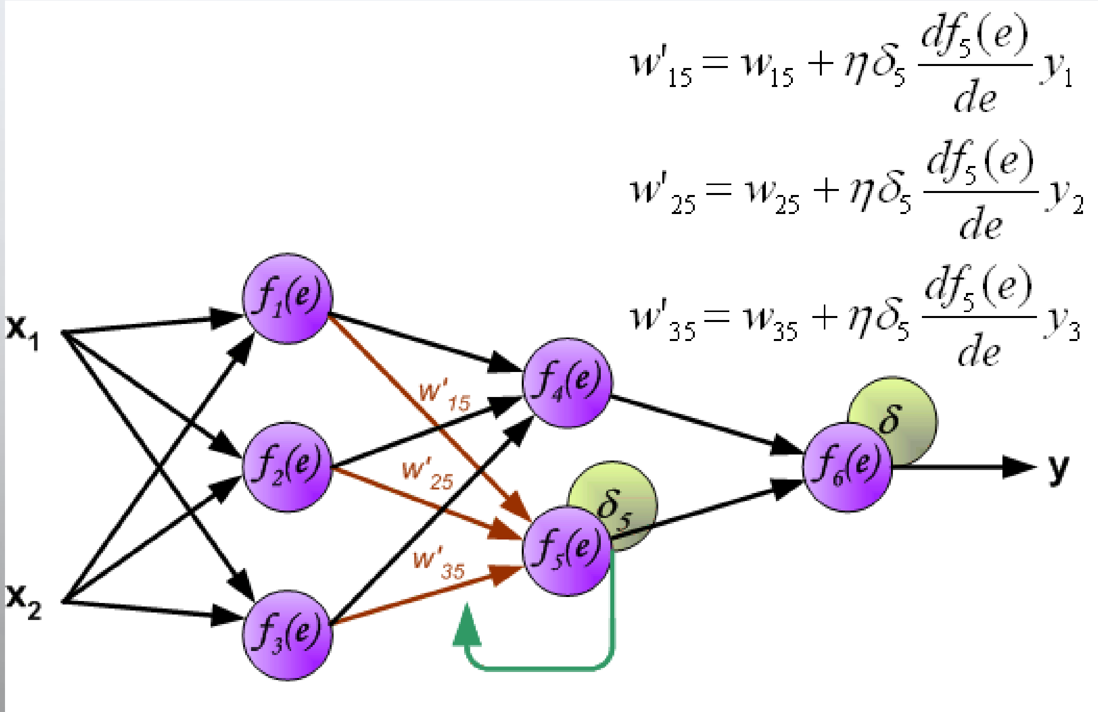
- Graph Partitioned and Distributed
- Send + Recv Replace Split Edges
- Send
  - Pushes value from one device to another
- Recv
  - Blocks until value available
- “Deferred execution”

EXECUTION

- Synchronous Execution
  - Classically frowned upon
  - GPUs make appealing
- All workers forced to take same parameters
- Backup workers stochastically eliminate straggling processes

DIFFERENTIATION & BACKPROP

- Symbolic representation
  - Automatically computes backprop code
- Like PS architectures, enables distributed training via +/- write operations



IMPLEMENTATION

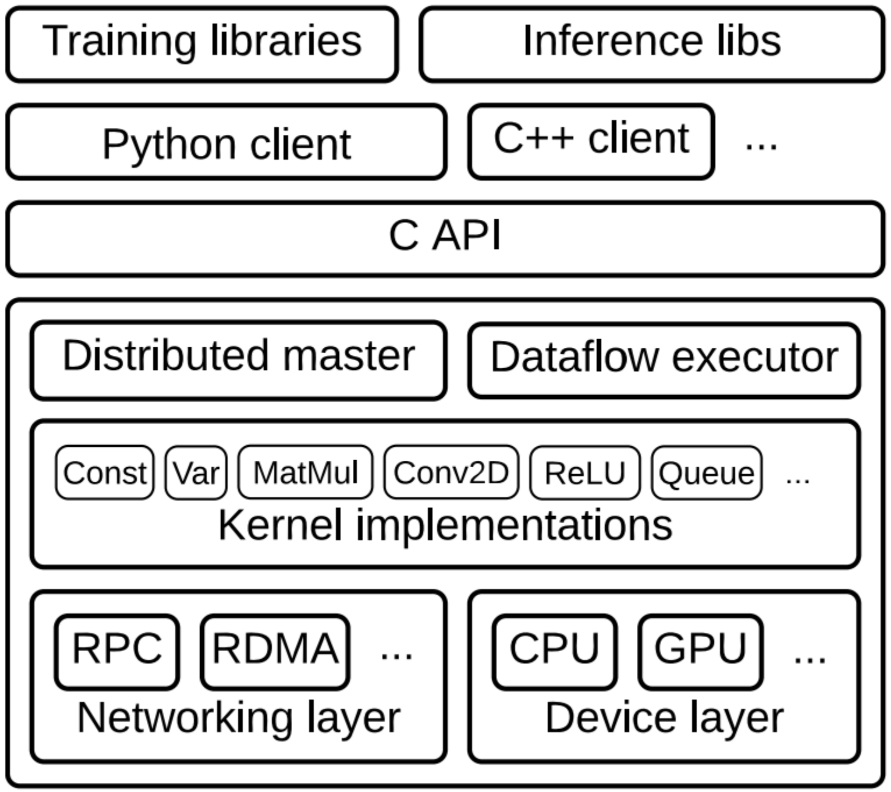


Figure 6: The layered TensorFlow architecture.

SINGLE MACHINE BENCHMARKS

Library	Training step time (ms)			
	AlexNet	Overfeat	OxfordNet	GoogleNet
Caffe [38]	324	823	1068	1935
Neon [58]	87	**211**	**320**	**270**
Torch [17]	**81**	268	529	470
TensorFlow	**81**	279	540	445

SPARSE AND DENSE FETCHES FOR SYNC

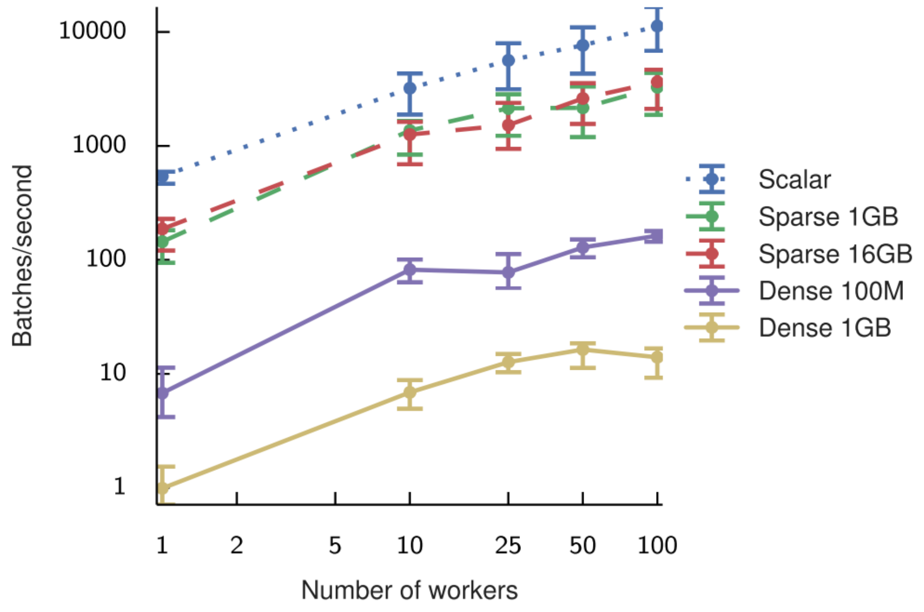


Figure 7: Baseline throughput for synchronous replication with a null model. Sparse accesses enable TensorFlow to handle larger models, such as embedding matrices (§4.2).

CNN IMPLEMENTATIONS

Library	Training step time (ms)			
	AlexNet	Overfeat	OxfordNet	GoogleNet
Caffe [38]	324	823	1068	1935
Neon [58]	87	**211**	**320**	**270**
Torch [17]	**81**	268	529	470
TensorFlow	**81**	279	540	445

Table 1: Step times for training four convolutional models with different libraries, using one GPU. All results are for training with 32-bit floats. The fastest time for each model is shown in bold.

SYNC AND NON-SYNCED PROCESSES

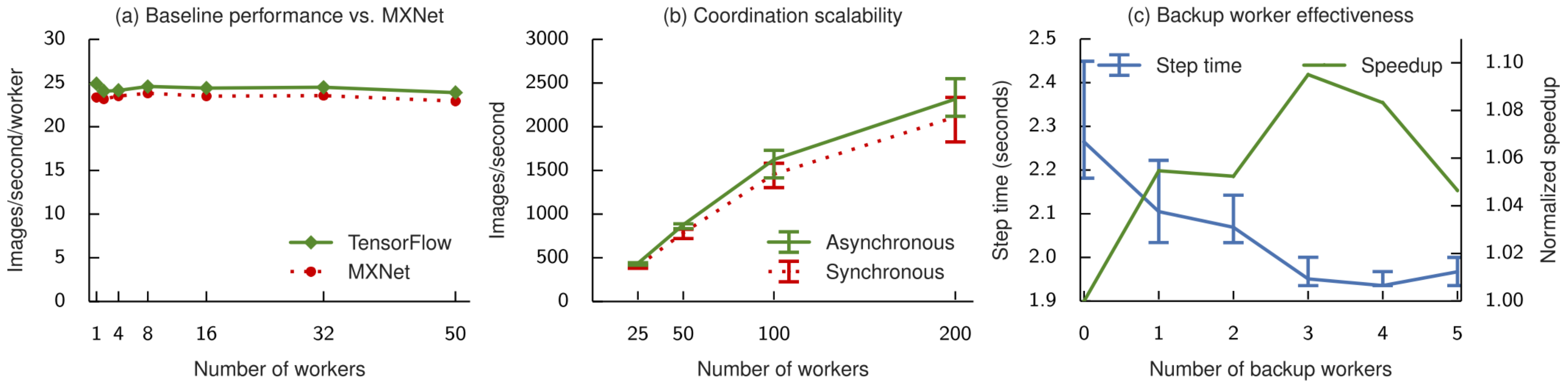


Figure 8: Results of the performance evaluation for Inception-v3 training (§6.3). (a) TensorFlow achieves slightly better throughput than MXNet for asynchronous training. (b) Asynchronous and synchronous training throughput increases with up to 200 workers. (c) Adding backup workers to a 50-worker training job can reduce the overall step time, and improve performance even when normalized for resource consumption.

TRAINING LARGE MODELS

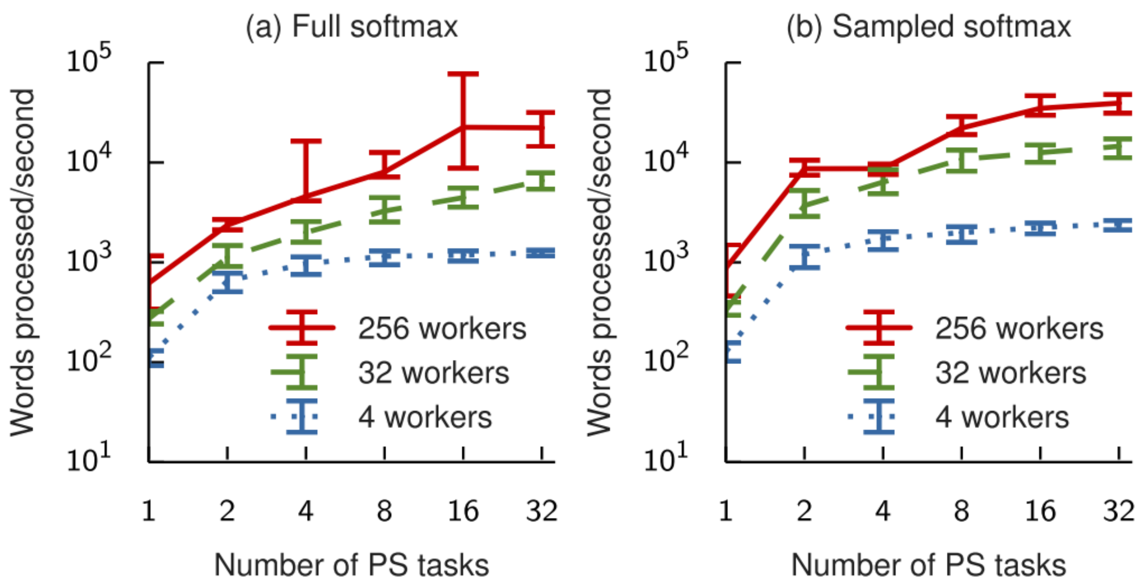


Figure 9: Increasing the number of PS tasks leads to increased throughput for language model training, by parallelizing the softmax computation. Sampled softmax increases throughput by performing less computation.

CRITICISM

- No actual accuracy comparisons
- Convergence comparisons in synchrony analysis?
- Lacking capability for abstracted computation
    - Reason why Keras runs on top of TF

CONCLUSION

- Built a ML system that is:
  - Robust
  - Distributable
  - Extensible
  - Fast
- In the ensuing years
  - Used extensively
  - Extended

REFERECES

- TensorFlow: A System for Large-Scale Machine Learning TensorFlow: A system for large-scale machine learning. *M. Abadi, P. Barham, J. Chen et al. 2016*