ADVANCES IN TRINITY OF AI: DATA, ALGORITHMS & COMPUTE

Anima Anandkumar

Bren Professor at Caltech
Director of ML Research at NVIDIA

TRINITY FUELING ARTIFICIAL INTELLIGENCE

ALGORITHMS

- OPTIMIZATION
- SCALABILITY
- MULTI-DIMENSIONALITY

INFRASTRUCTURE

FULL STACK FOR ML

- APPLICATION SERVICES
- ML PLATFORM
- GPUS

DATA

- COLLECTION
- AGGREGATION
- AUGMENTATION

DATA

- **COLLECTION:** ACTIVE LEARNING, PARTIAL LABELS..
- **AGGREGATION:** CROWDSOURCING MODELS..
- **AUGMENTATION:** GENERATIVE MODELS, SYMBOLIC EXPRESSIONS..

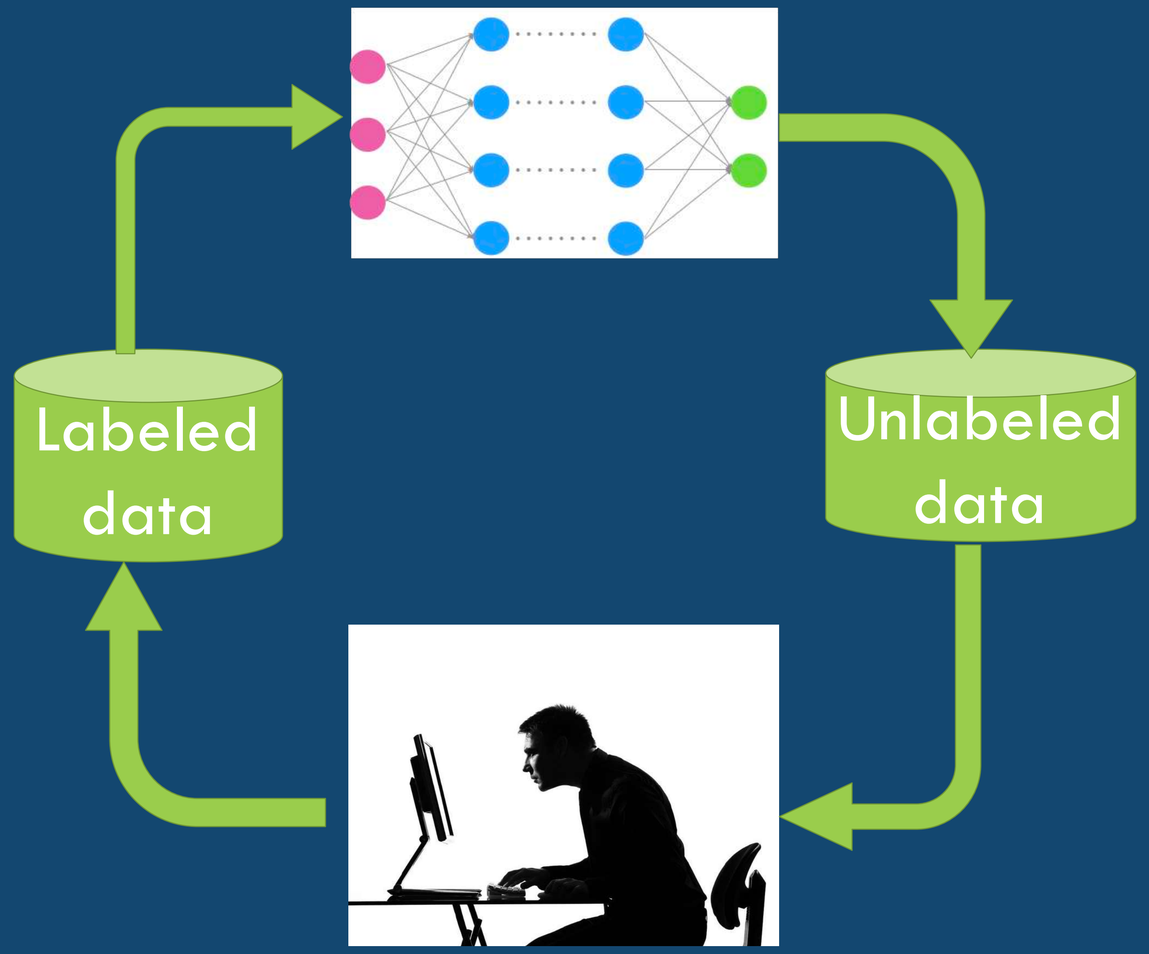
ACTIVE LEARNING

Goal

- Reach SOTA with a smaller dataset

- Active learning analyzed in theory
- In practice, only small classical models

Can it work at scale with deep learning?



TASK: NAMED ENTITY RECOGNITION

In 1917, Einstein applied the general theory of relativity to model the large-scale structure of the universe. He was visiting the United States when Adolf Hitler came to power in 1933 and did not go back to Germany, where he had been a professor at the Berlin Academy of Sciences. He settled in the U.S., becoming an American citizen in 1940. On the eve of World War II, he endorsed a letter to President Franklin D. Roosevelt alerting him to the potential development of "extremely powerful bombs of a new type" and recommending that the U.S. begin similar research. This eventually led to what would become the Manhattan Project. Einstein supported defending the Allied forces, but largely denounced using the new discovery of nuclear fission as a weapon. Later, with the British philosopher Bertrand Russell, Einstein signed the Russell-Einstein Manifesto, which highlighted the danger of nuclear weapons. Einstein was affiliated with the Institute for Advanced Study in Princeton, New Jersey, until his death in 1955.

Tag colours:

LOCATION TIME PERSON ORGANIZATION MONEY PERCENT DATE

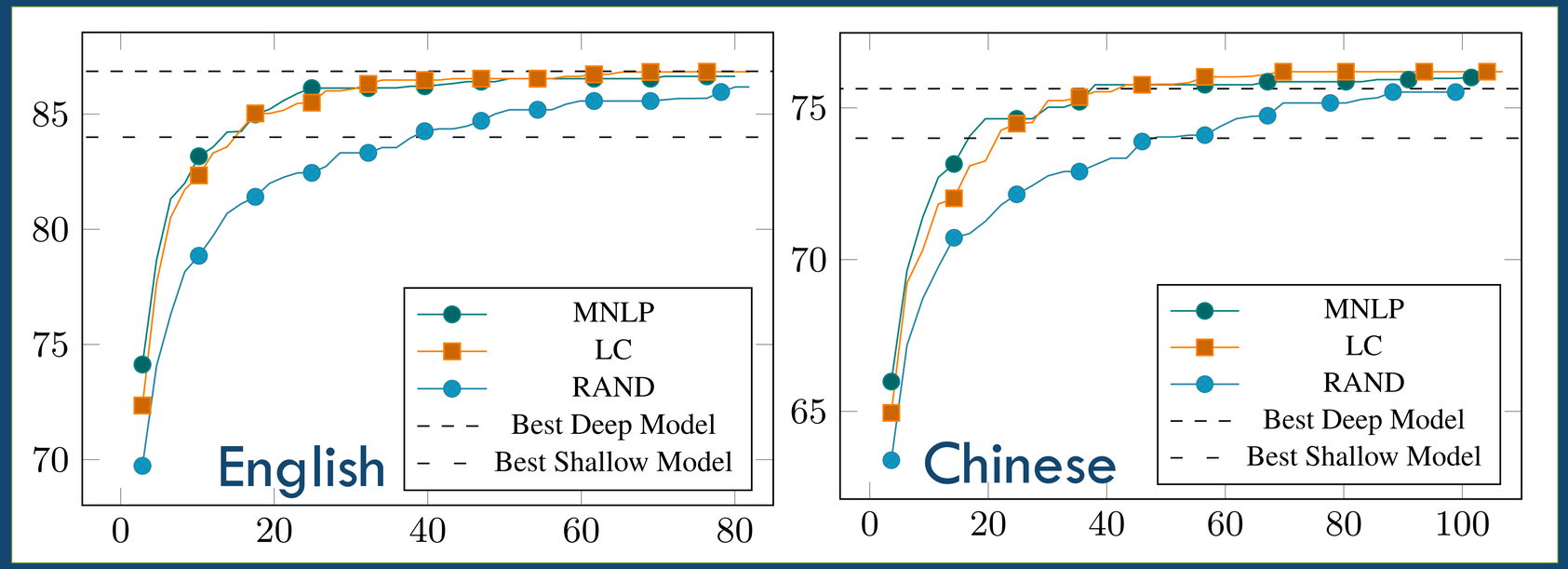
RESULTS

NER task on largest open benchmark (Onto-notes)

Test F1 score vs. % of labeled words

Active learning heuristics:

- Least confidence (LC)
- Max. normalized log probability (MNLP)

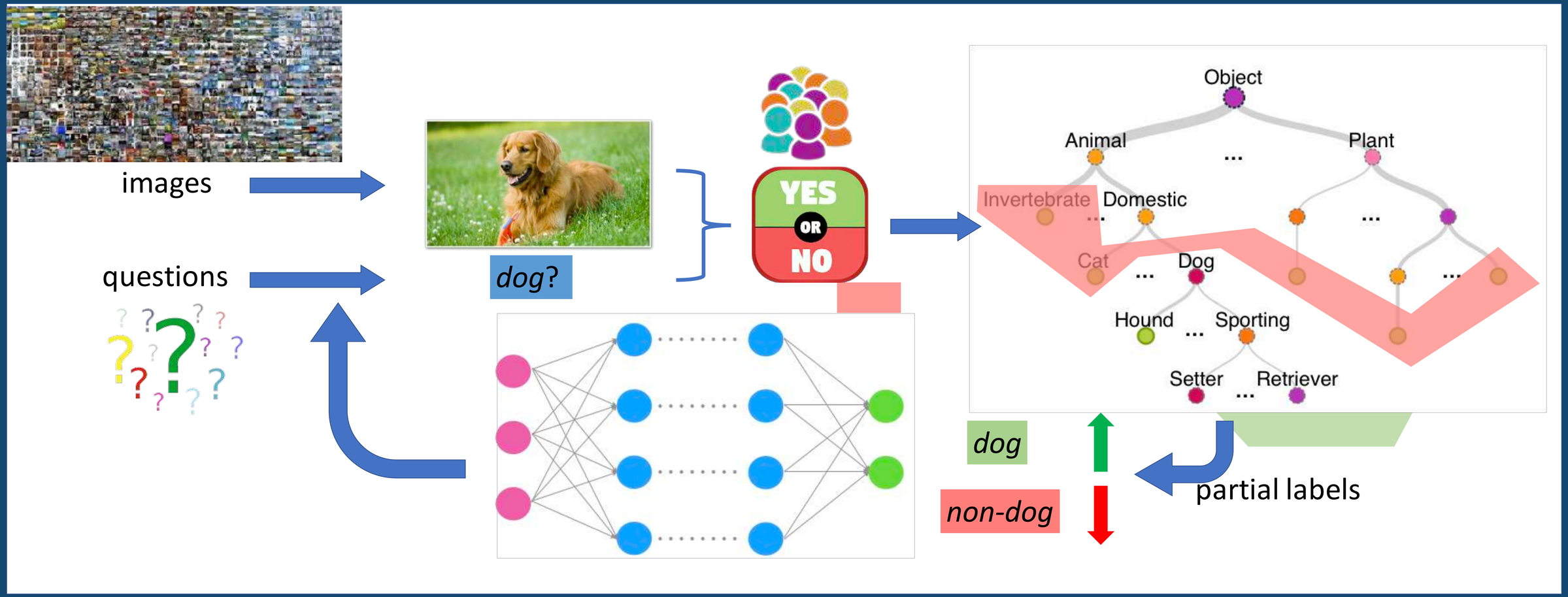


- Deep active learning matches :
  - SOTA with just **25%** data on English, **30%** on Chinese.
  - Best shallow model (on full data) with **12%** data on English, **17%** on Chinese.

TAKE-AWAY

- Uncertainty sampling works. Normalizing for length helps under low data.
- With active learning, **deep beats shallow** even in low data regime.
- With active learning, **SOTA achieved** with far fewer samples.

ACTIVE LEARNING WITH PARTIAL FEEDBACK

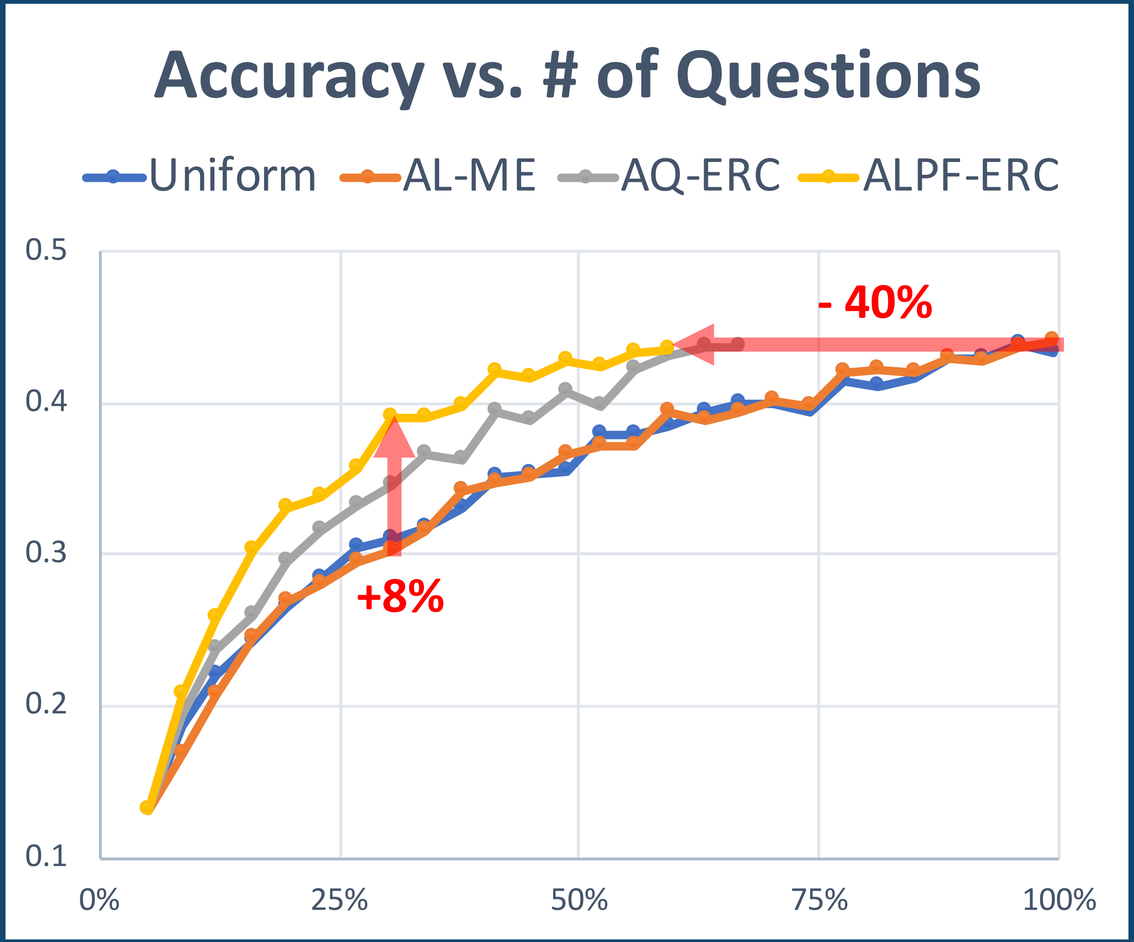


- Hierarchical class labeling: Labor proportional to # of binary questions asked
- **Actively pick informative questions ?**

RESULTS ON TINY IMAGENET (100K SAMPLES)

ALPF-ERC
***active** data*
***active** questions*

AQ-ERC
***inactive** data*
***active** questions*



Uniform
***inactive** data*
***inactive** questions*

AL-ME
***active** data*
***inactive** questions*

- Yield **8%** higher accuracy at **30%** questions (w.r.t. Uniform)
- Obtain full annotation with **40%** less binary questions

TWO TAKE-AWAYS

- Don’t annotate from scratch
  - Select questions actively based on the learned model
- Don’t sleep on partial labels
  - Re-train model from partial labels

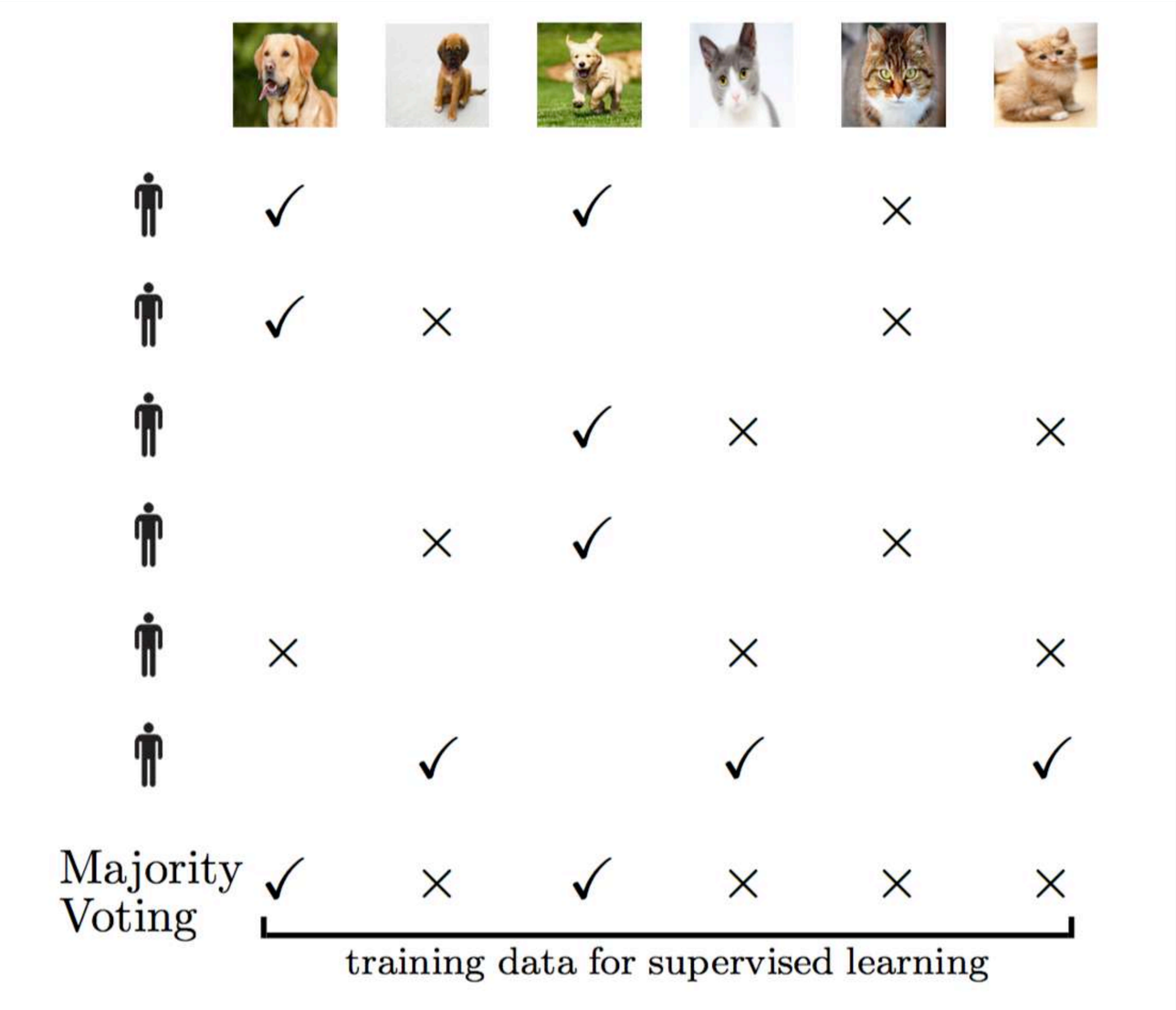
CROWDSOURCING: AGGREGATION OF CROWD ANNOTATIONS

Majority rule

- Simple and common.
- Wasteful: ignores annotator quality of different workers.

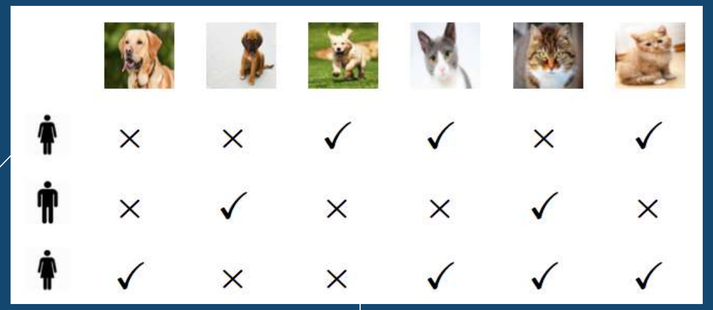
Annotator-quality models

- Can improve accuracy.
- Hard: needs to be estimated without ground-truth.



PROPOSED CROWDSOURCING ALGORITHM

Noisy crowdsourced annotations

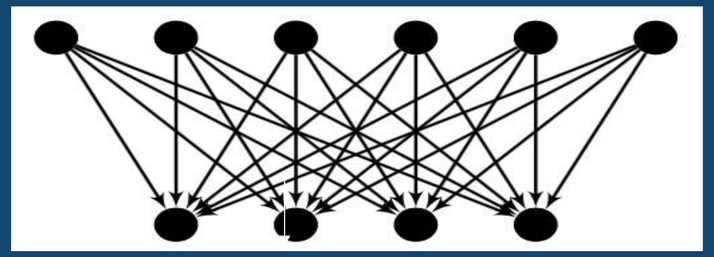


Repeat

Posterior of ground-truth labels given annotator quality model

cat	1/3	1/3	1/3	2/3	2/3	2/3
not cat	2/3	2/3	2/3	1/3	1/3	1/3

Training with weighted loss. Use posterior as weights



Use trained model to infer ground-truth labels

cat	0	0	1	1	1	0
not cat	1	1	0	0	0	1

MLE : update Annotator quality using inferred labels from model

0.7	0.3	0.5

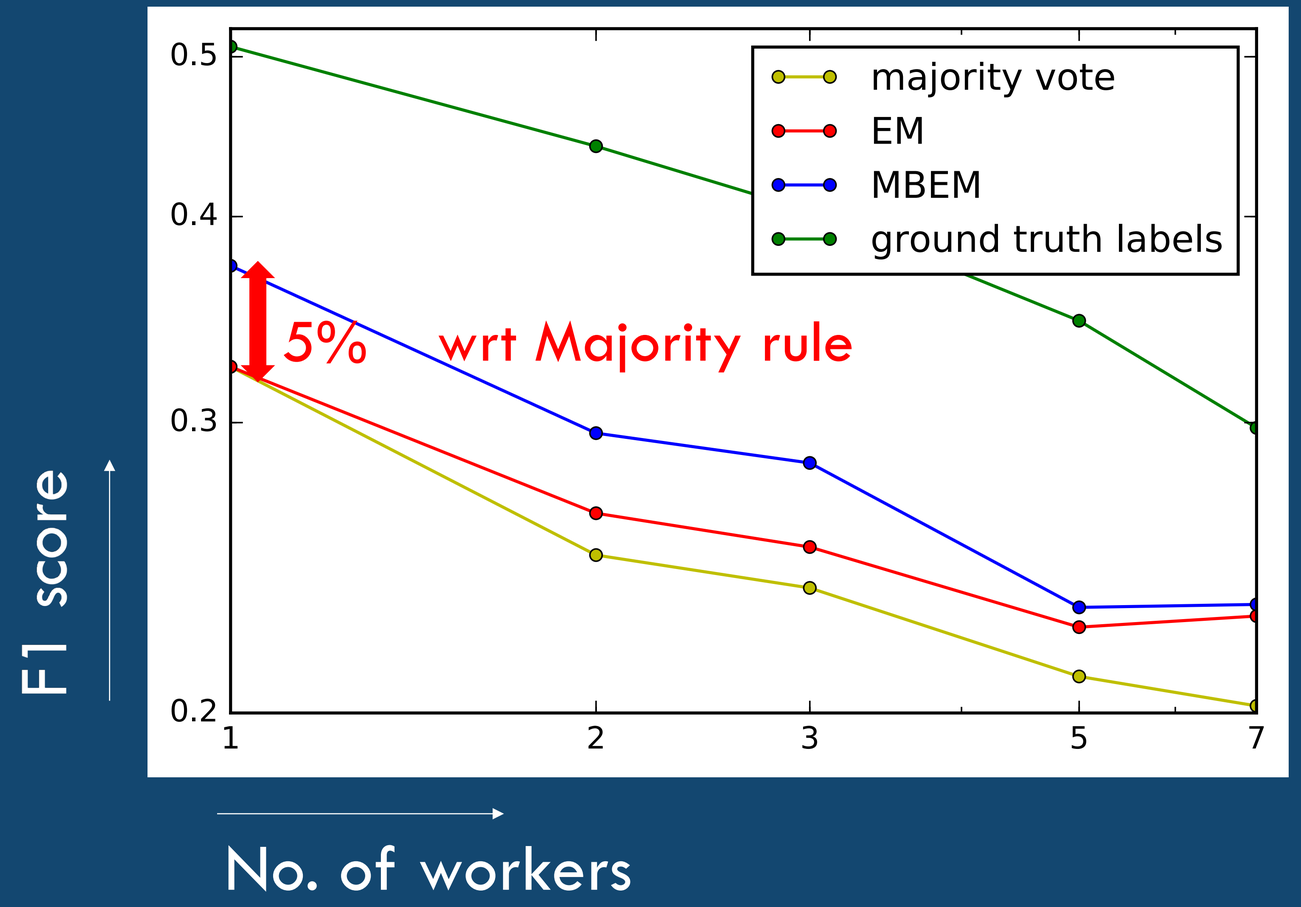
LABELING ONCE IS OPTIMAL: BOTH IN THEORY AND PRACTICE

Theorem: Under fixed budget, generalization error minimized with **single annotation** per sample.

Assumptions:

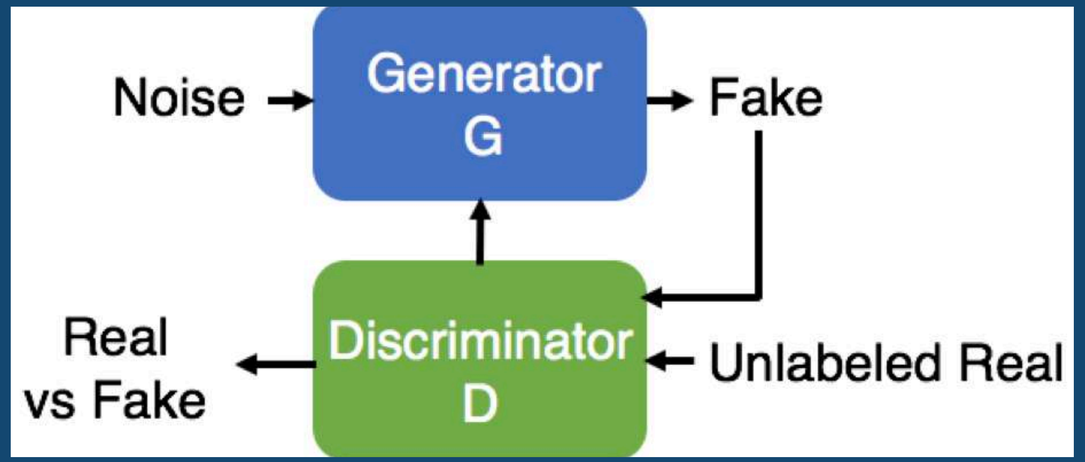
- Best predictor is accurate enough (under no label noise).
- Simplified case: All workers have same quality.
- Prob. of being correct > 83%

MS-COCO dataset. Fixed budget: 35k annotations



DATA AUGMENTATION 1: GENERATIVE MODELING

GAN



Merits

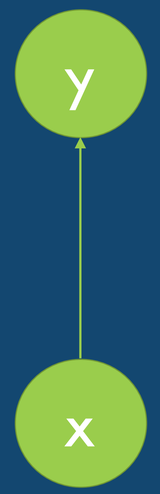
- Captures statistics of natural images
- Learnable

Peril

- Feedback is real vs. fake: different from prediction.
- Introduces artifacts

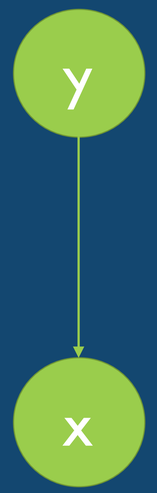
PREDICTIVE VS GENERATIVE MODELS

$P(y \mid x)$



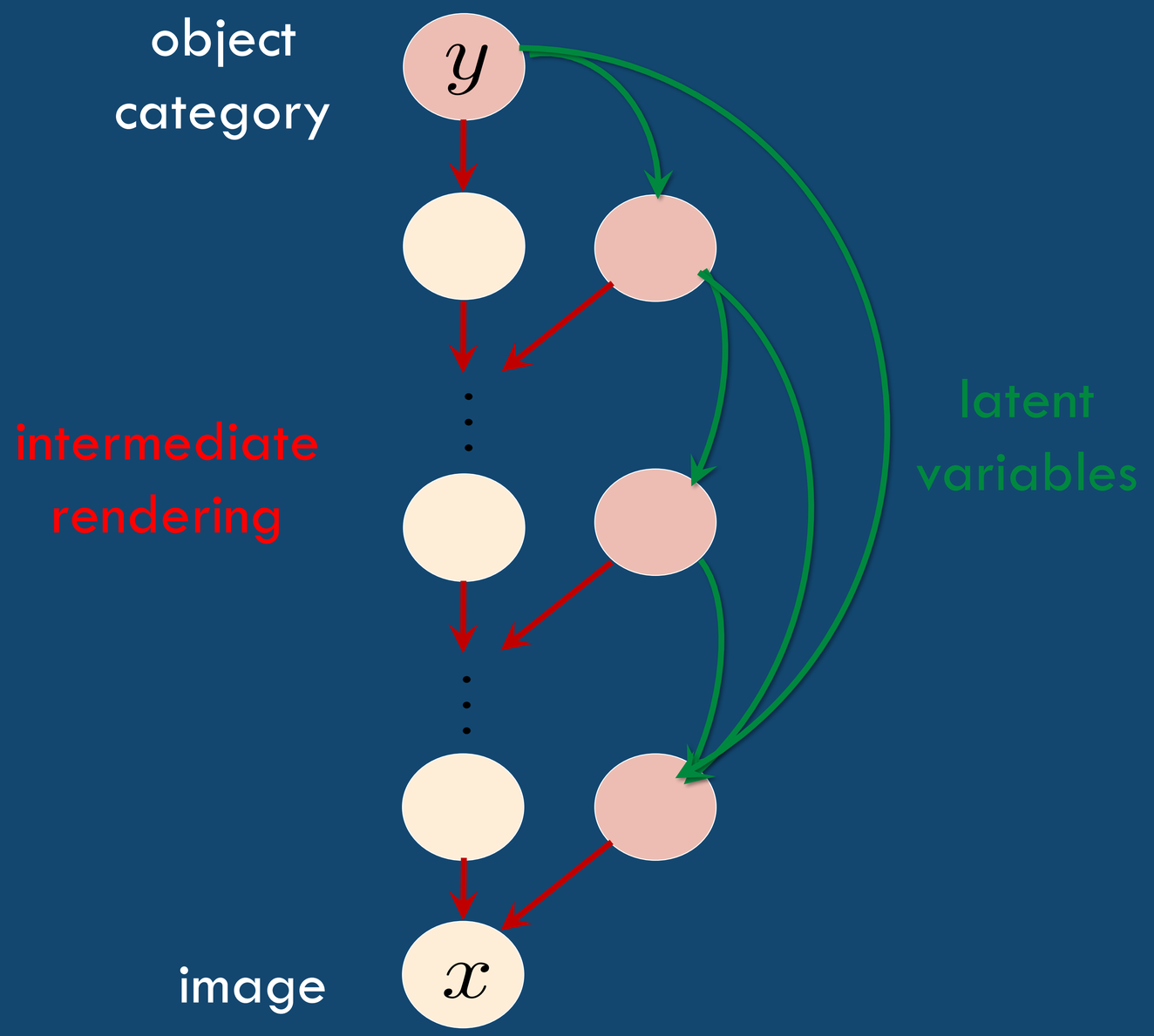
One model to do both?

$P(x \mid y)$



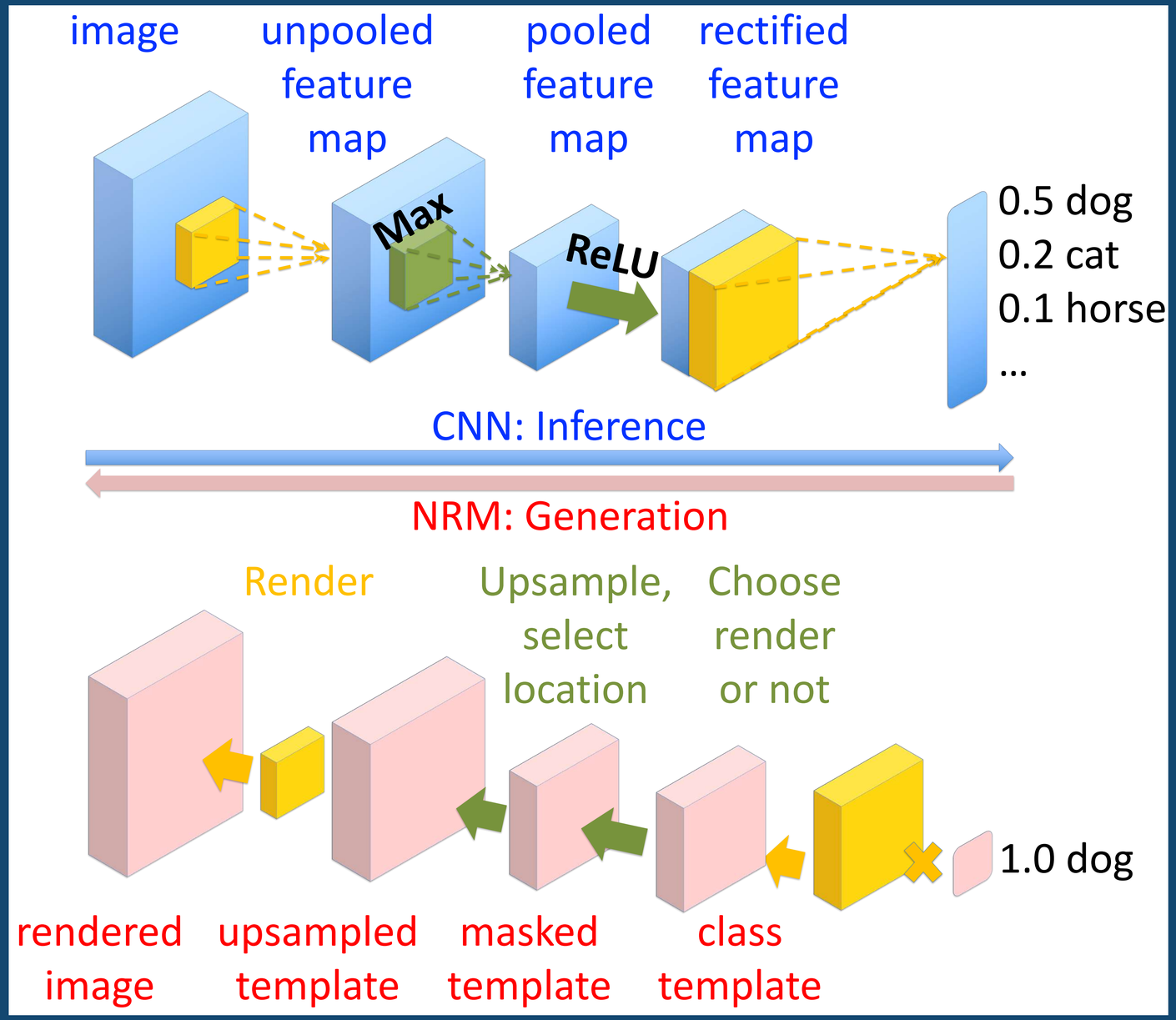
- SOTA prediction from CNN models.
- What class of $p(x|y)$ yield CNN models for $p(y|x)$?

NEURAL DEEP RENDERING MODEL (NRM)



Design joint priors for latent variables based on reverse-engineering CNN predictive architectures

NEURAL RENDERING MODEL (NRM)



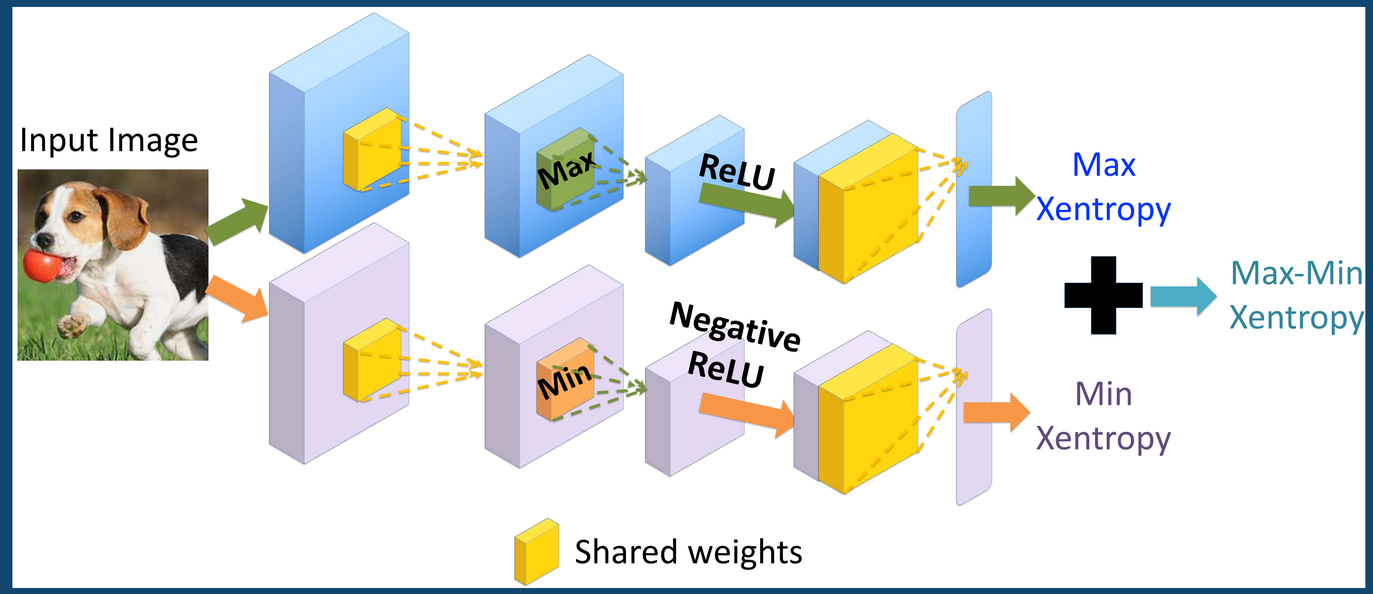
MAX-MIN CROSS-ENTROPY ➡ MAX-MIN NETWORKS

Cross-Entropy Loss for Training the CNNs with Labeled Data

$$\min_{\theta \in \mathcal{A}_\gamma} H_{p,q}(y|x, z^{\max}) \geq \min_{(z_i)_{i=1}^n, \theta} \frac{1}{n} \sum_{i=1}^{n} -\log p(y_i|x_i, z_i; \theta)$$

Max-Min Loss for Training the CNNs with Labeled Data

$$\alpha^{\max} H_{p,q}(y|x, z^{\max}) + \alpha^{\min} H_{p,q}(y|x, z^{\min})$$



- Max cross-entropy maximizes the posteriors of correct labels.
  Min cross-entropy minimizes the posteriors of incorrect labels.
- Co-learning: Max and Min networks try to learn from each other

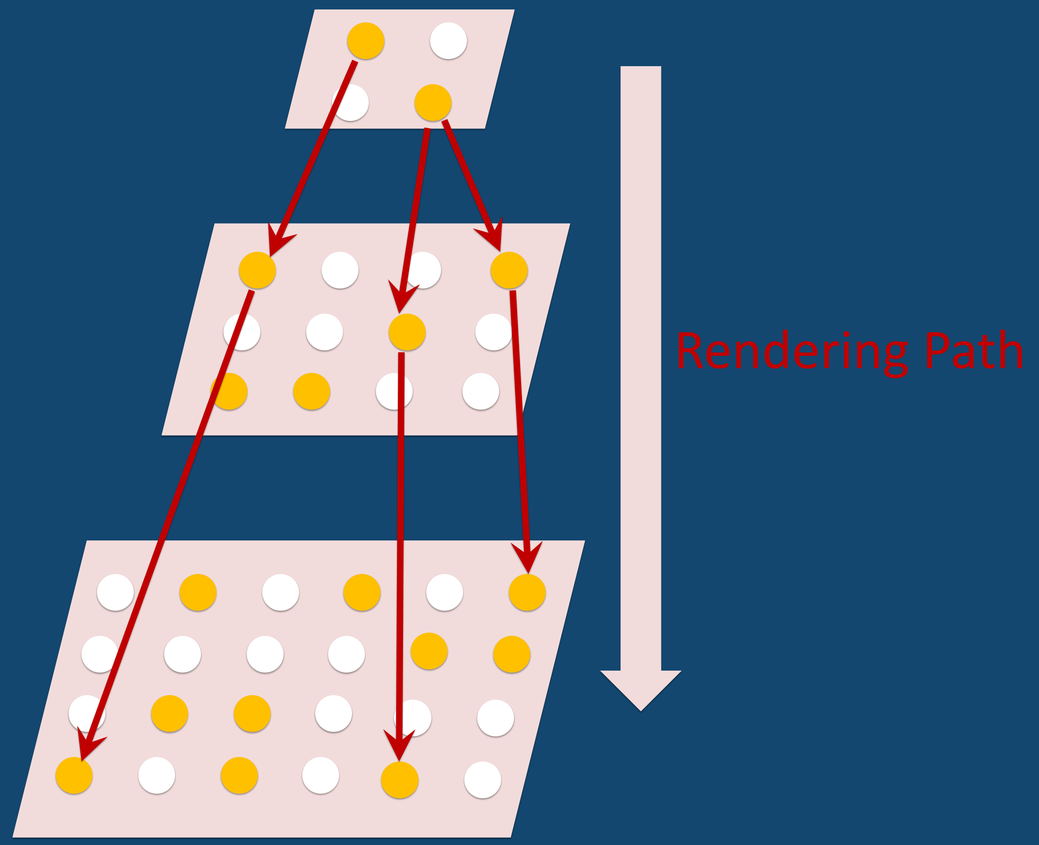
STATISTICAL GUARANTEES FOR THE NRM

Training loss in the CNNs equivalent to likelihood in NRM

Bound on the generalization error

Risk $\leq$

$$\frac{Number\ of\ active\ rendering\ \ paths}{n^{1/2}}$$



- Rendering path normalization:
- new form of regularization

Max-Min NRM with RPN achieves SOTA on benchmarks

DATA AUGMENTATION 2: SYMBOLIC EXPRESSIONS

Goal: Learn a domain of functions (sin, cos, log, add…)

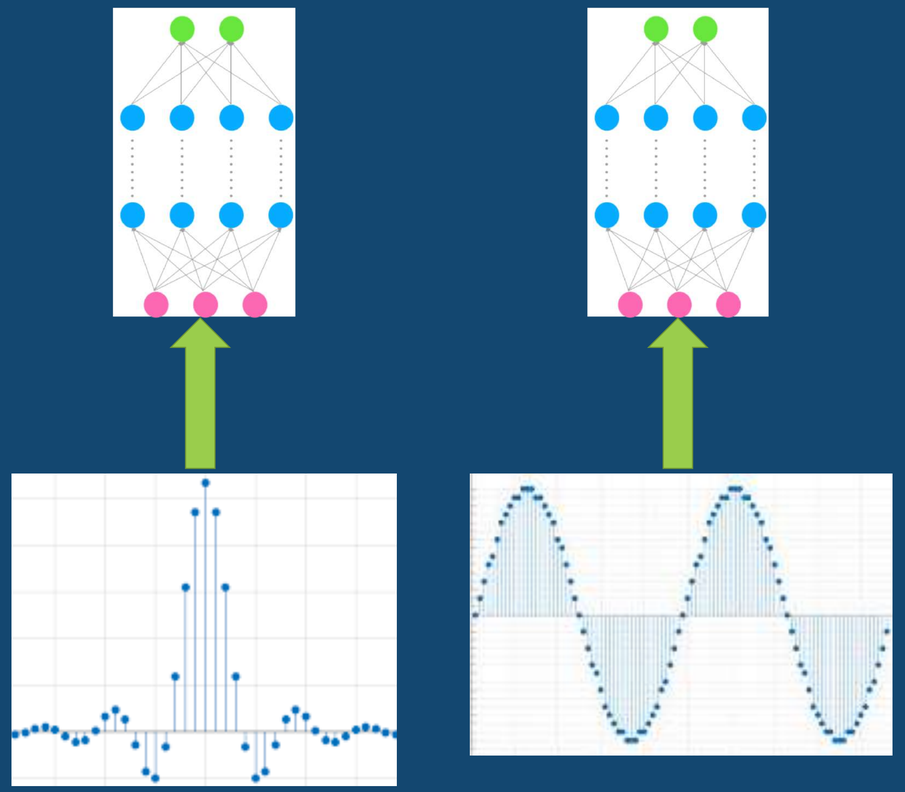
- Training on numerical input-output does not generalize.

Data Augmentation with Symbolic Expressions

- Efficiently encode relationships between functions.

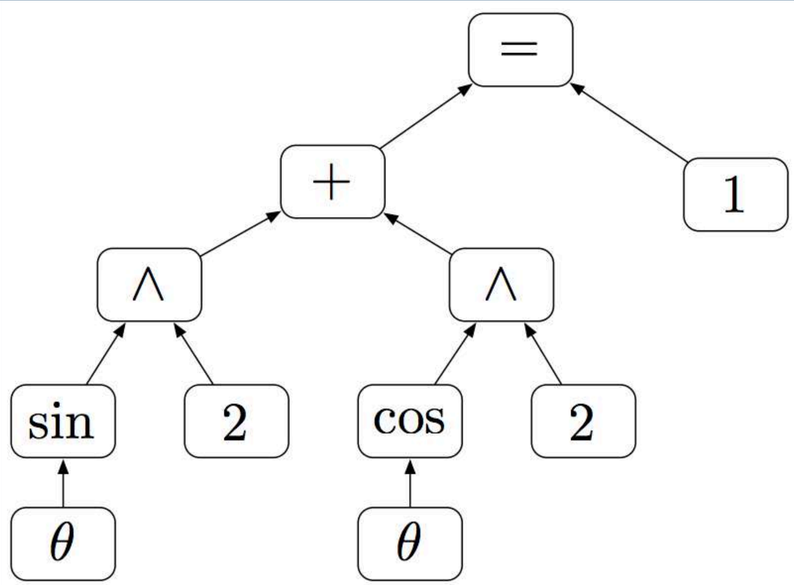
Solution:

- Design networks to use both: **symbolic + numerical**

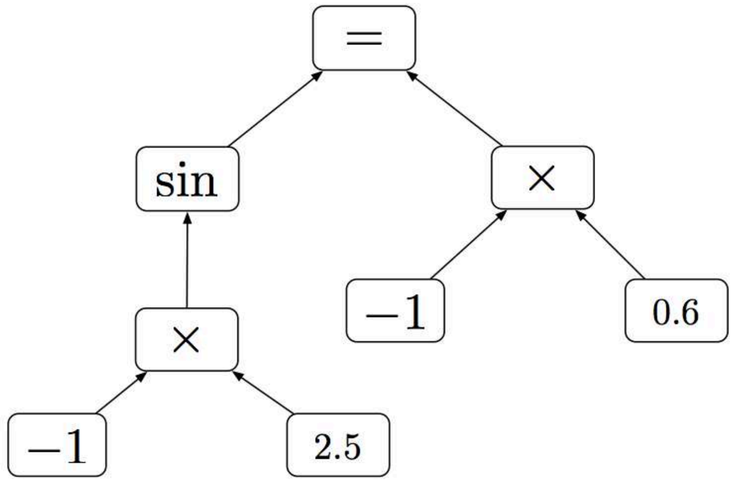


ARCHITECTURE : TREE LSTM

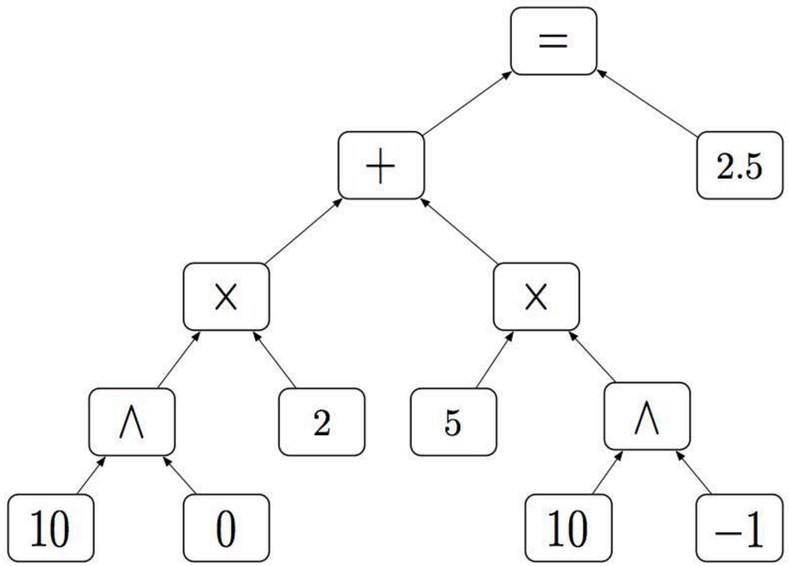
- Symbolic expression trees. Function evaluation tree.
- Decimal trees: encode numbers with decimal representation (numerical).
- Can encode any expression, function evaluation and number.



$\sin^2(\theta) + \cos^2(\theta) = 1$



$\sin(-2.5) = -0.6$



Decimal Tree for 2.5

RESULTS

- Vastly Improved numerical evaluation: 90% over function-fitting baseline.
- Generalization to verifying symbolic equations of higher depth

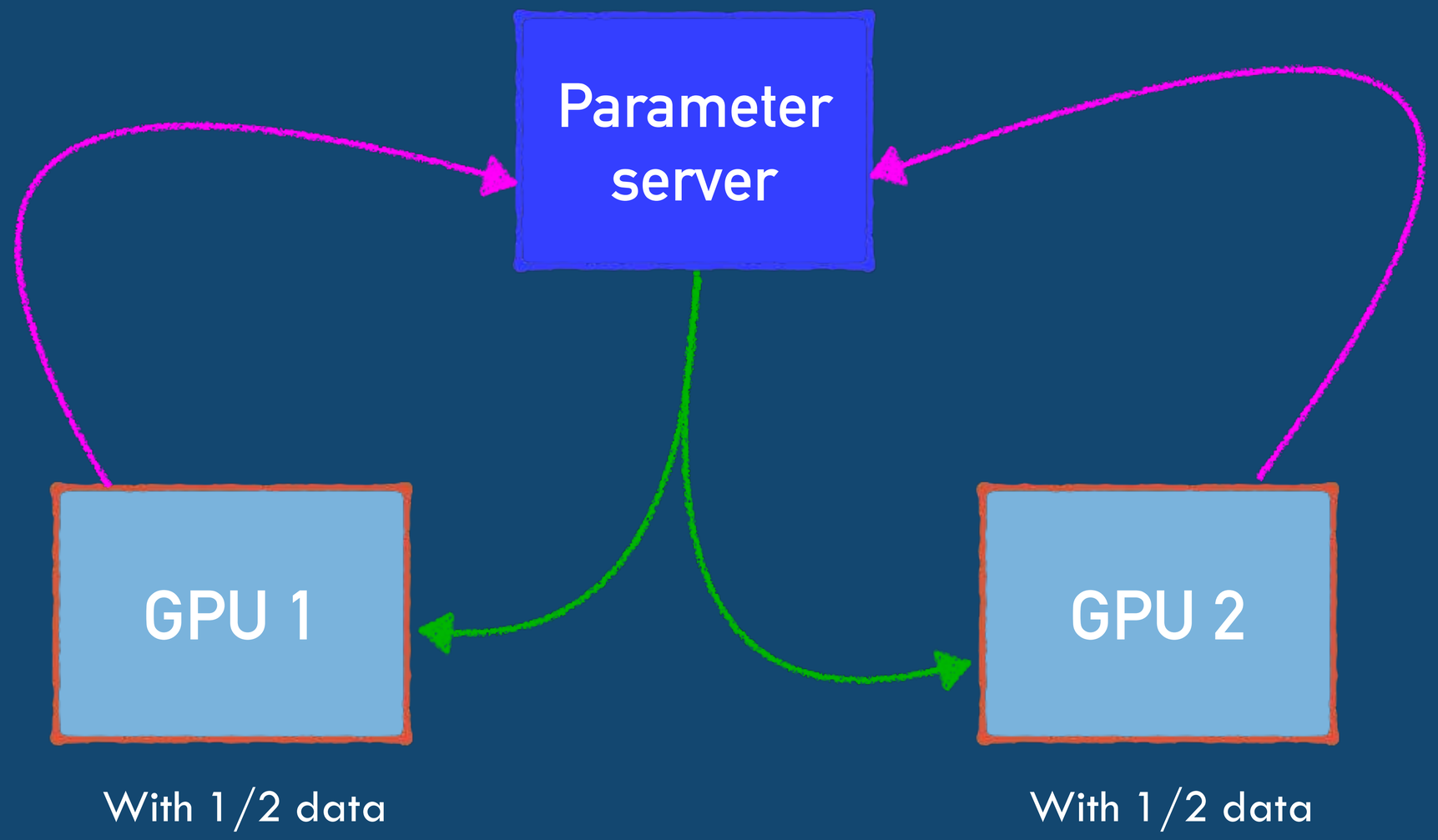
LSTM: Symbolic	TreeLSTM: Symbolic	TreeLSTM: symbolic + numeric
76.40 %	93.27 %	**96.17 %**

- **Combining symbolic + numerical data helps in better generalization for both tasks: symbolic and numerical evaluation.**

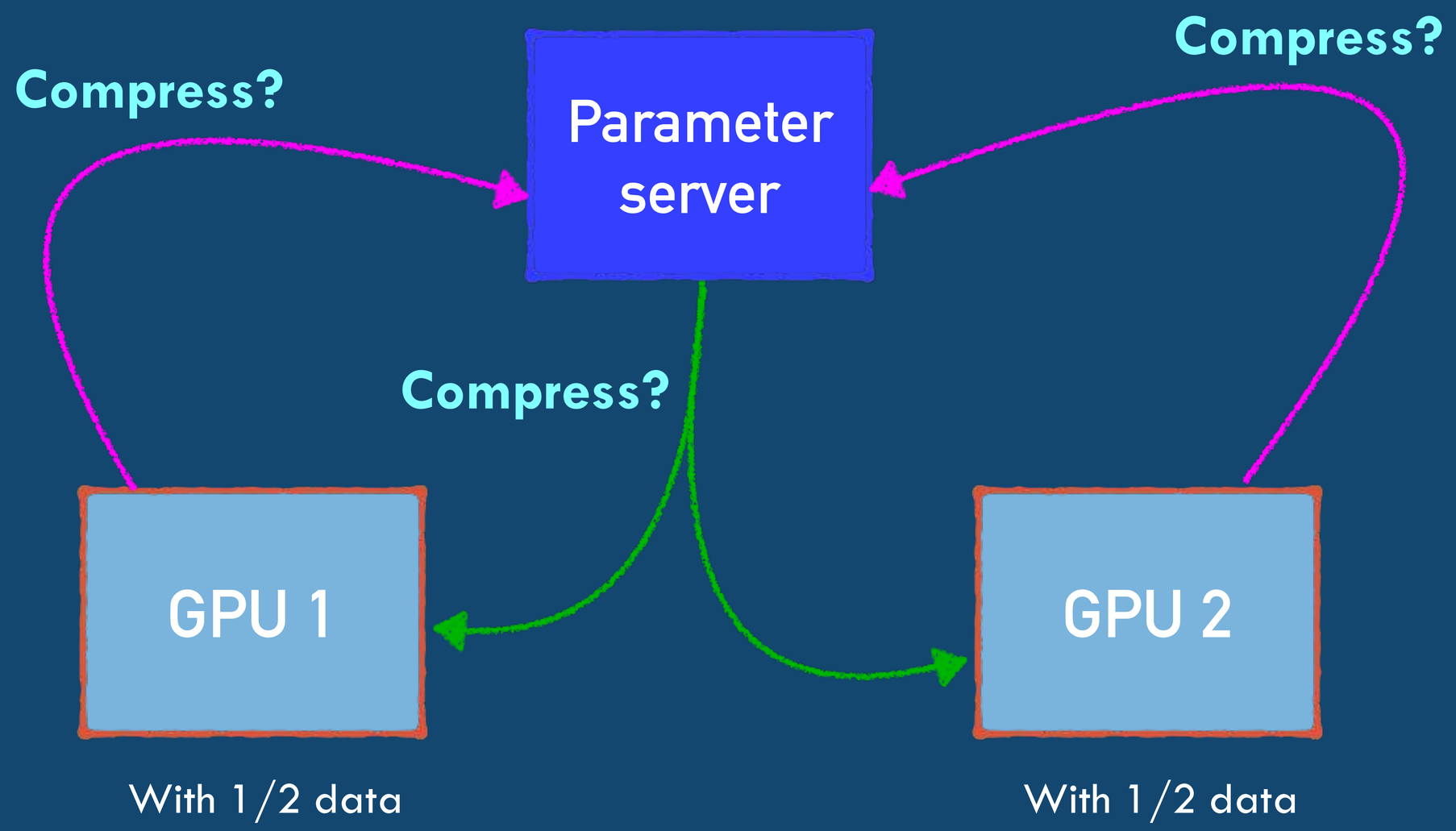
ALGORITHMS

- OPTIMIZATION : ANALYSIS OF CONVERGENCE
- SCALABILITY : GRADIENT QUANTIZATION
- MULTI-DIMENSIONALITY : TENSOR ALGEBRA

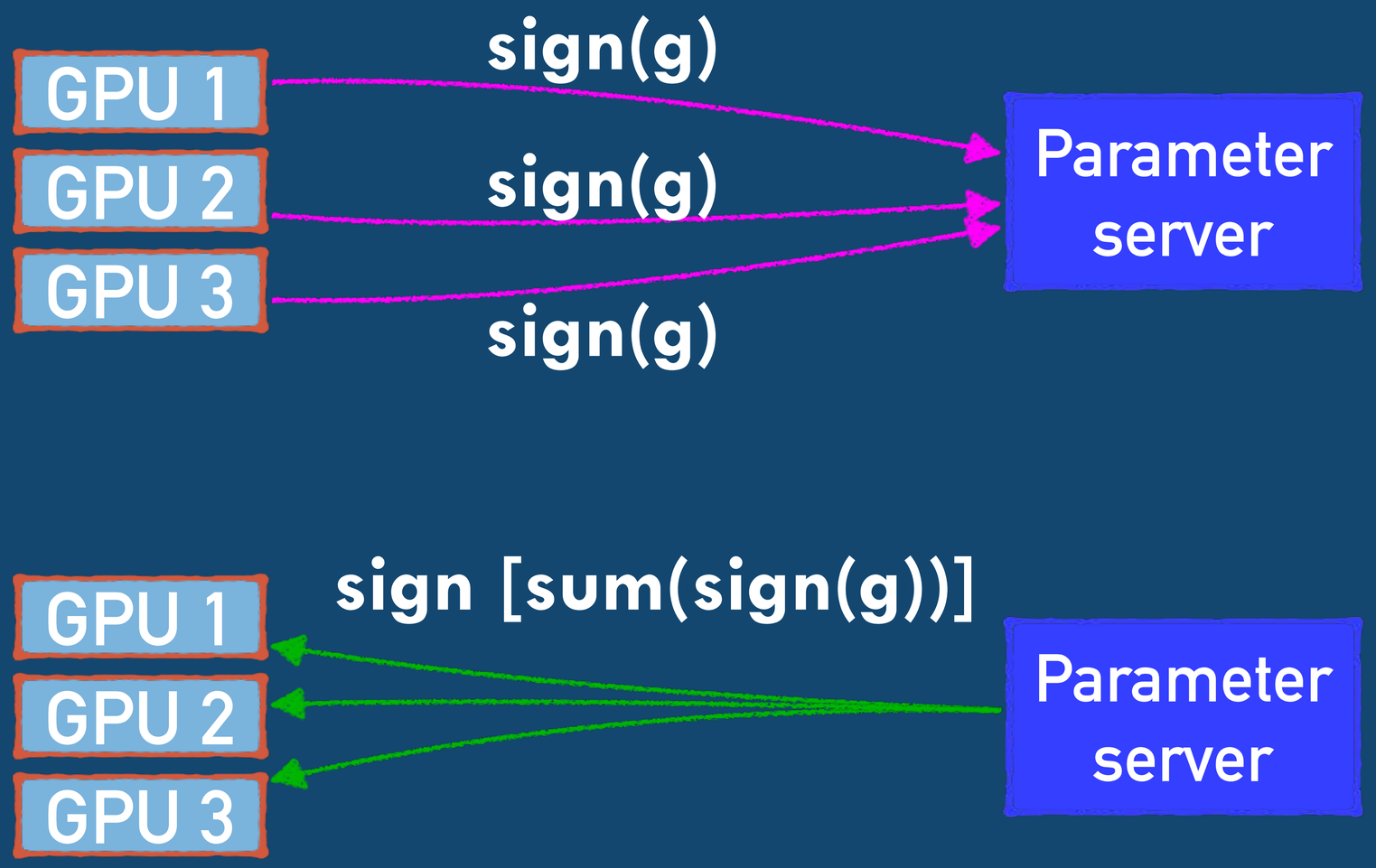
DISTRIBUTED TRAINING INVOLVES COMPUTATION & COMMUNICATION



DISTRIBUTED TRAINING INVOLVES COMPUTATION & COMMUNICATION

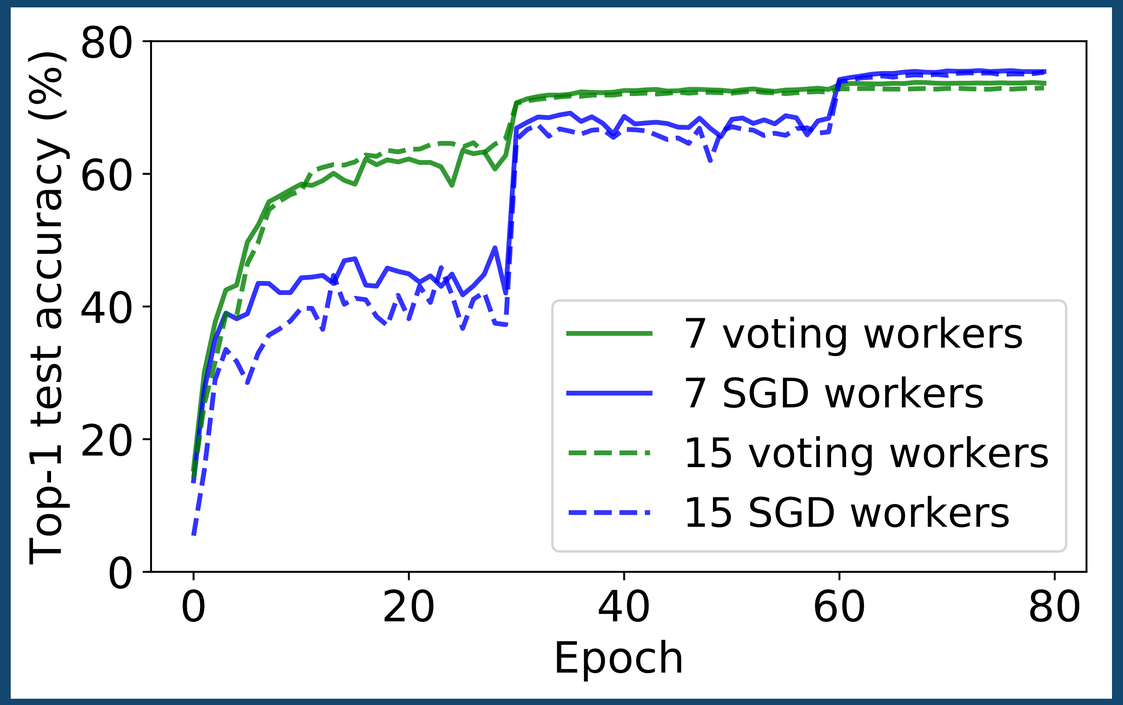


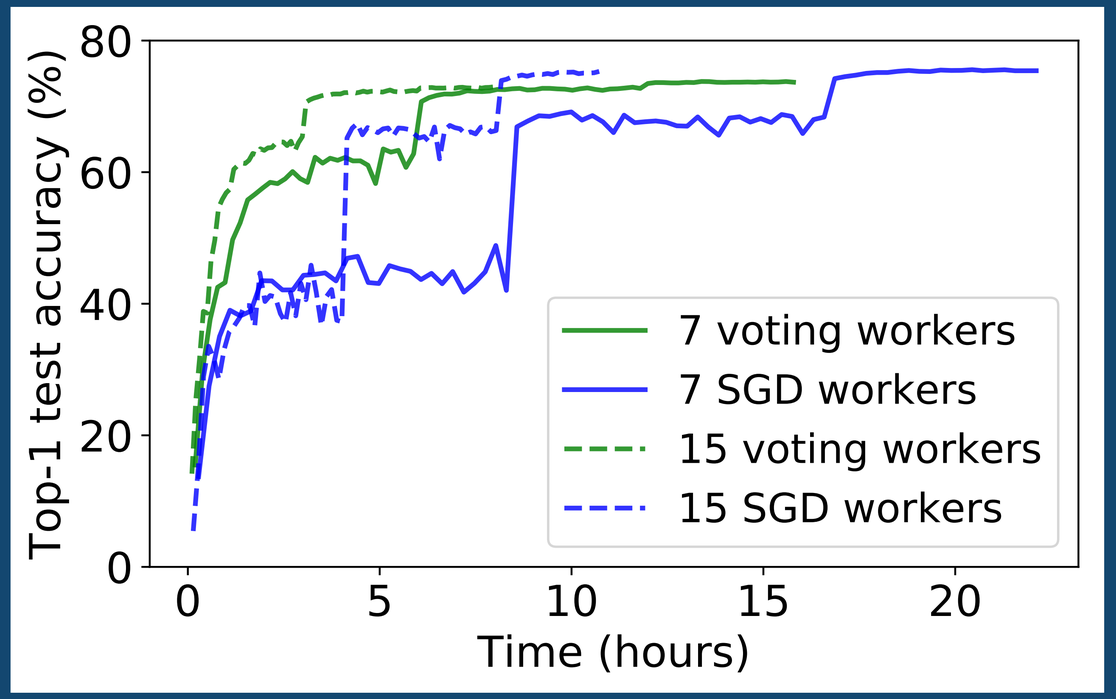
DISTRIBUTED TRAINING BY MAJORITY VOTE



SIGNSGD PROVIDES “FREE LUNCH"

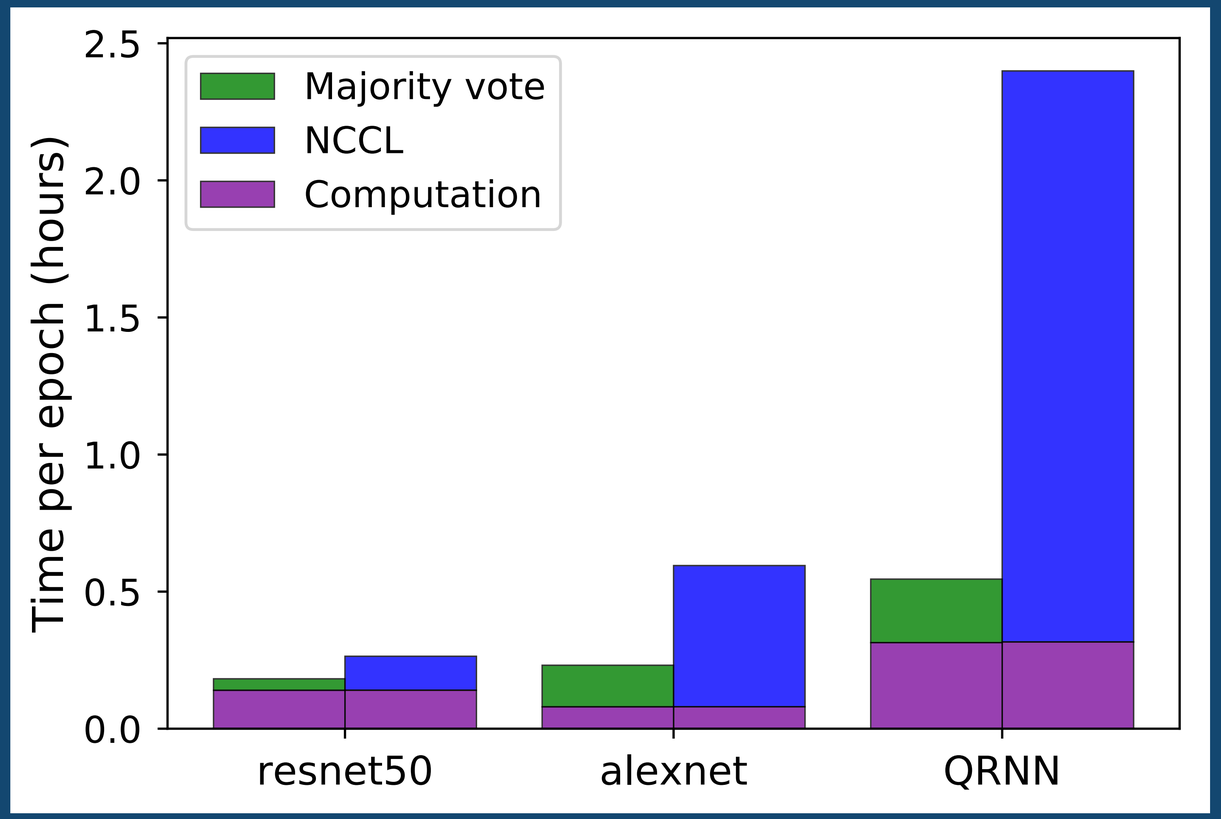
P3.2x machines on AWS, Resnet50 on imagenet





Throughput gain with only tiny accuracy loss

SIGNSGD ACROSS DOMAINS AND ARCHITECTURES

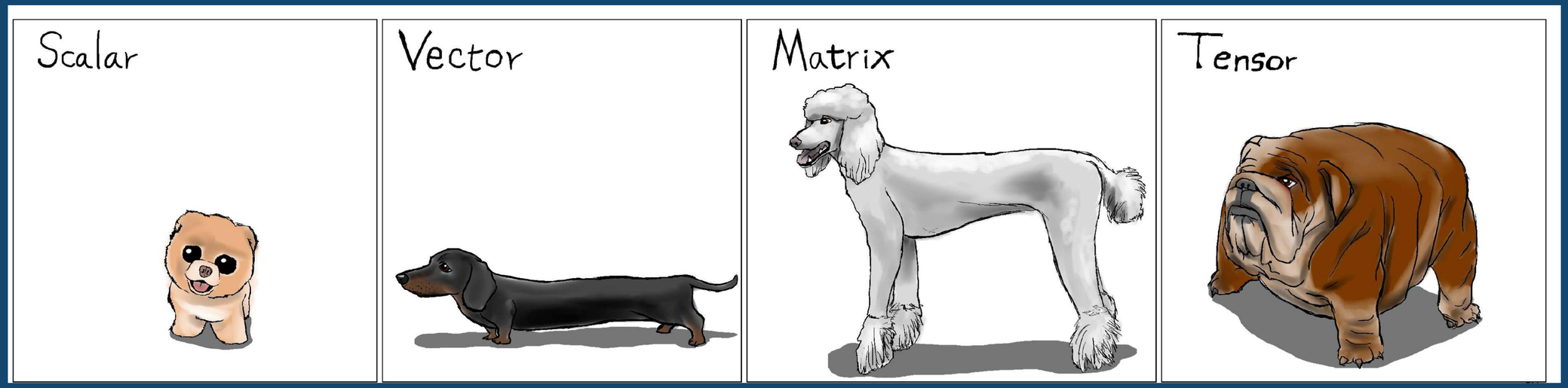


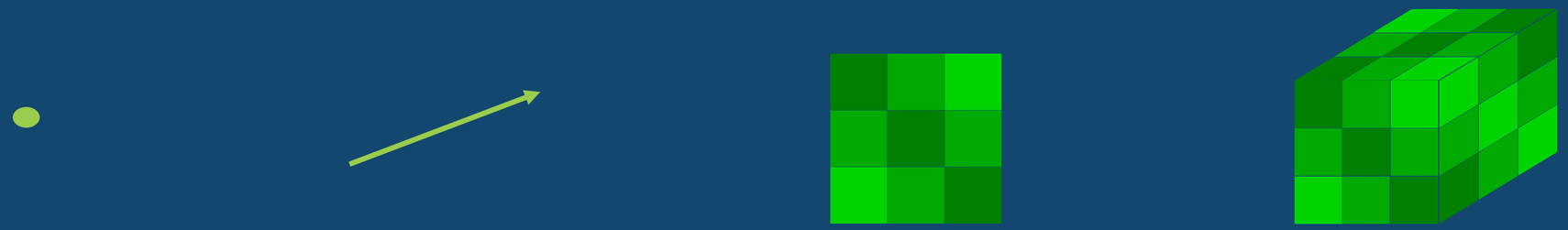
Huge throughput gain!

TAKE-AWAYS FOR SIGN-SGD

- Convergence even under biased gradients and noise.
- **Faster than SGD** in theory and in practice.
- For distributed training, similar variance reduction as SGD.
- In practice, similar accuracy but with **far less communication.**

TENSORS FOR LEARNING IN MANY DIMENSIONS

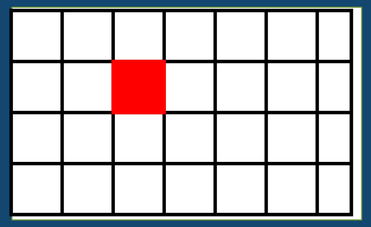




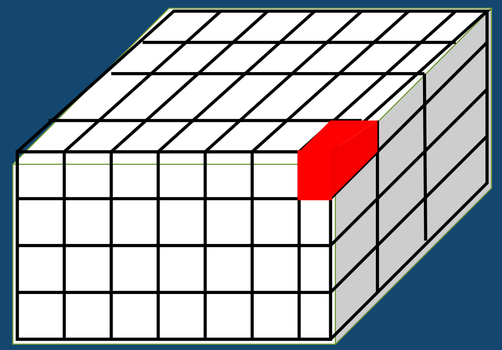
TENSORS FOR MULTI-DIMENSIONAL DATA AND HIGHER ORDER MOMENTS

Images: 3 dimensions

Videos: 4 dimensions



Pairwise correlations



Triplet correlations

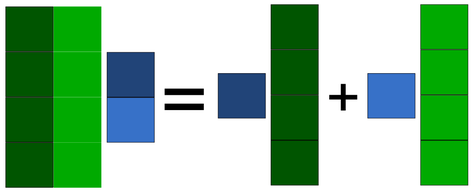
OPERATIONS ON TENSORS: TENSOR CONTRACTION

Tensor Contraction

Extends the notion of matrix product

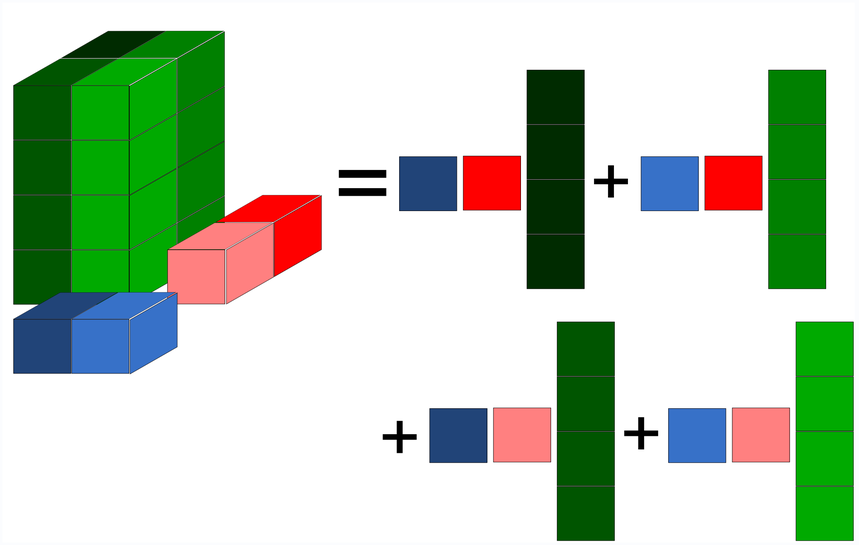
Matrix product

$$Mv = \sum_j v_j M_j$$

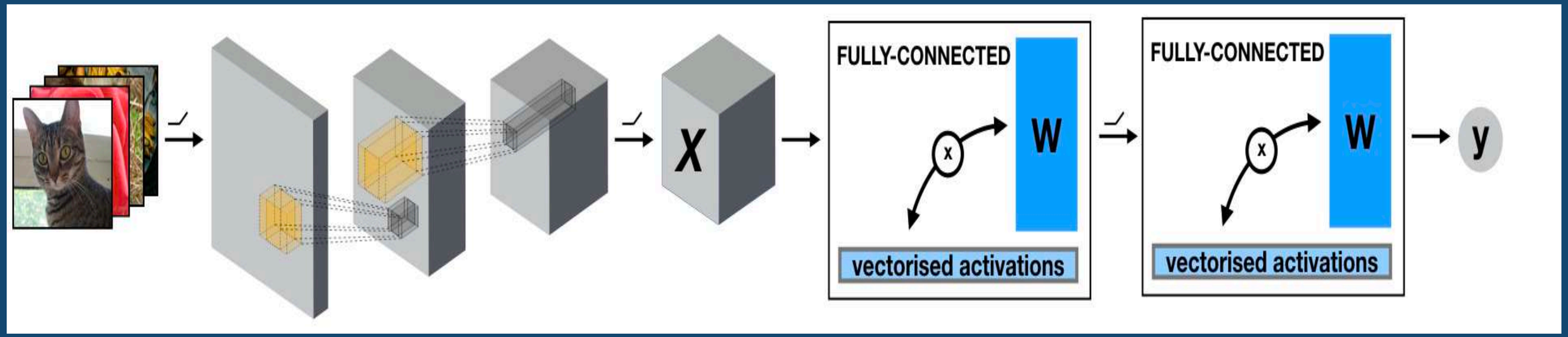


Tensor Contraction

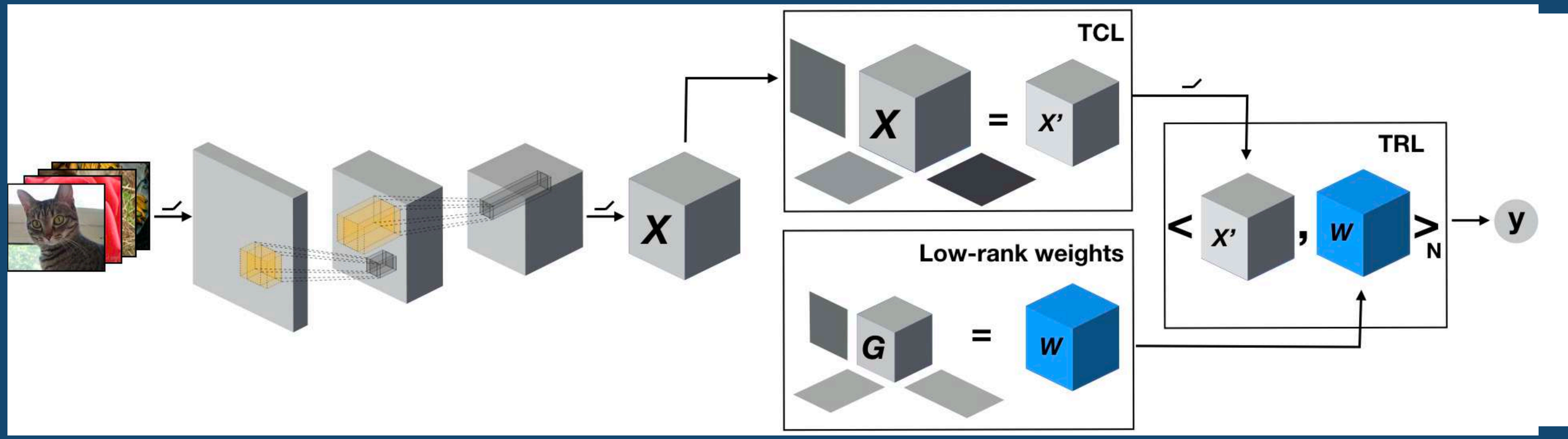
$$T(u, v, \cdot) = \sum_{i,j} u_i v_j T_{i,j,:}$$



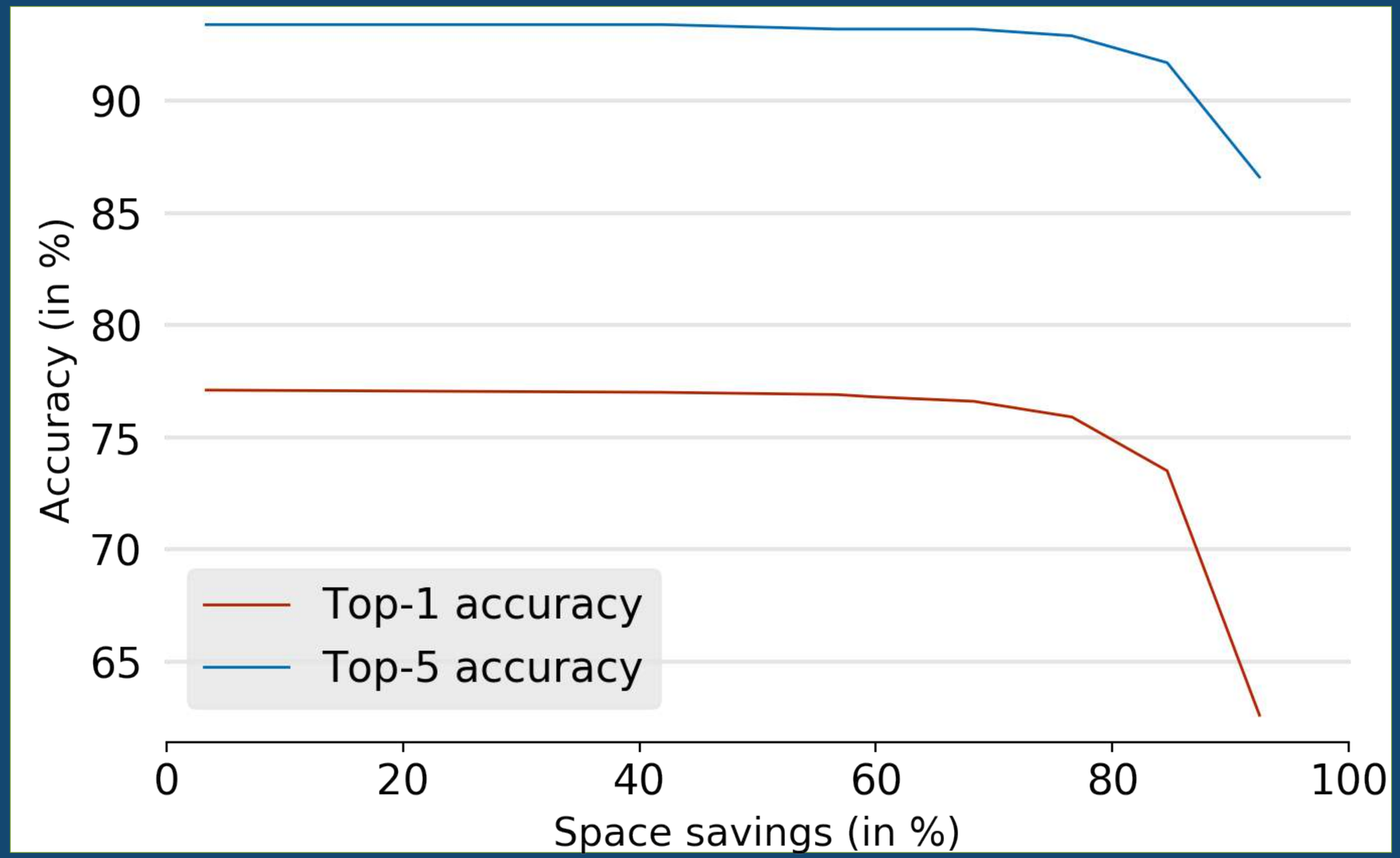
DEEP NEURAL NETS: TRANSFORMING TENSORS



DEEP TENSORIZED NETWORKS



SPACE SAVING IN DEEP TENSORIZED NETWORKS

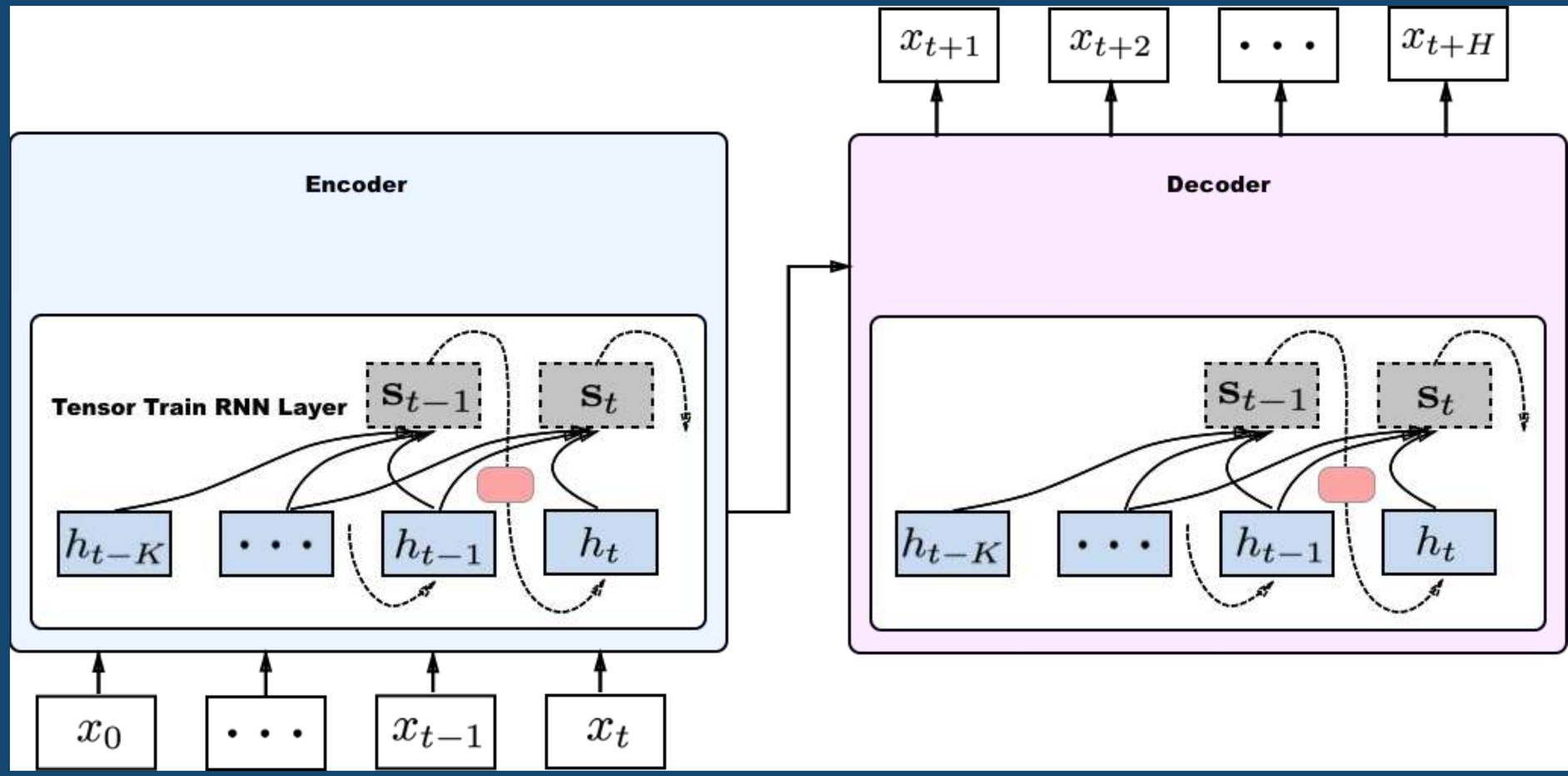


TENSORS FOR LONG-TERM FORECASTING

Tensor Train RNN and LSTMs

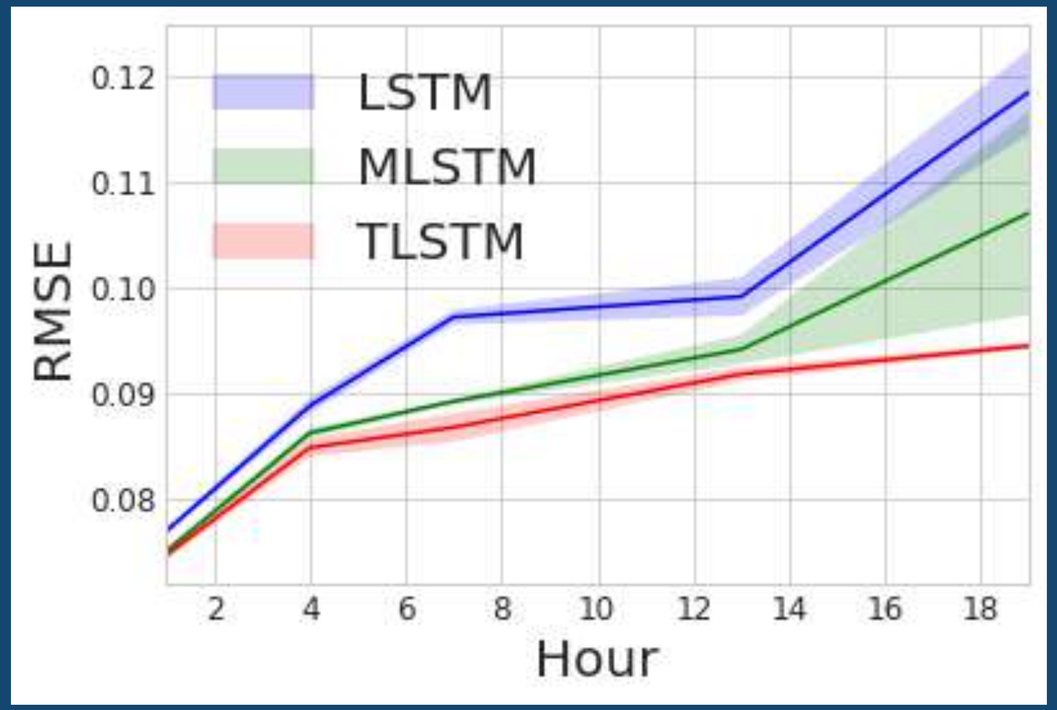
Challenges:

- Long-term dependencies
- High-order correlations
- Error propagation

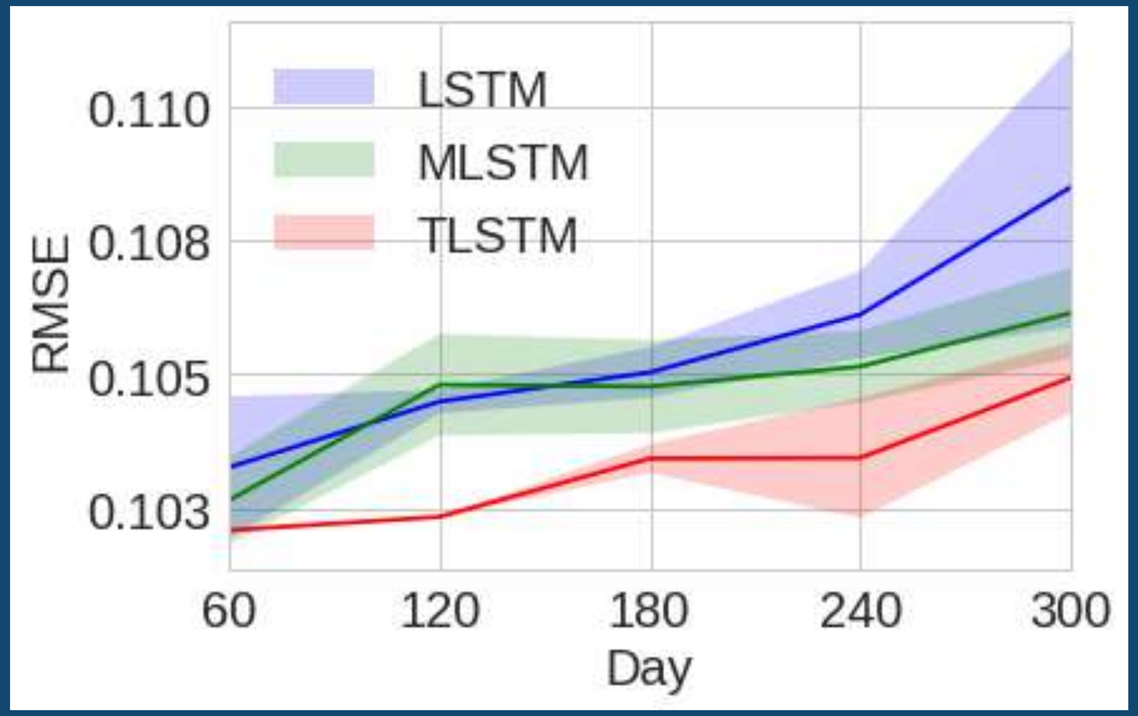


TENSOR LSTM FOR LONG-TERM FORECASTING

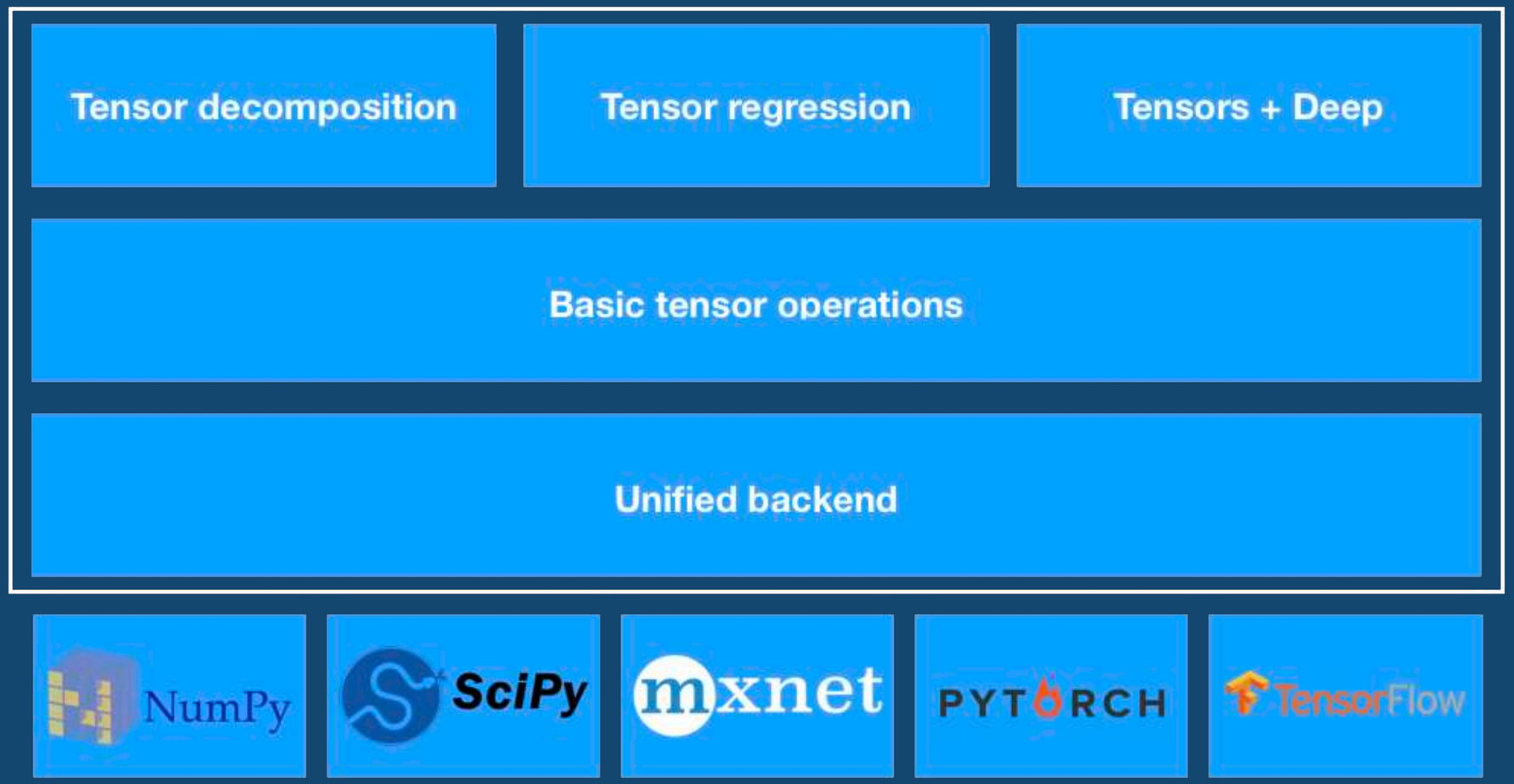
Traffic dataset



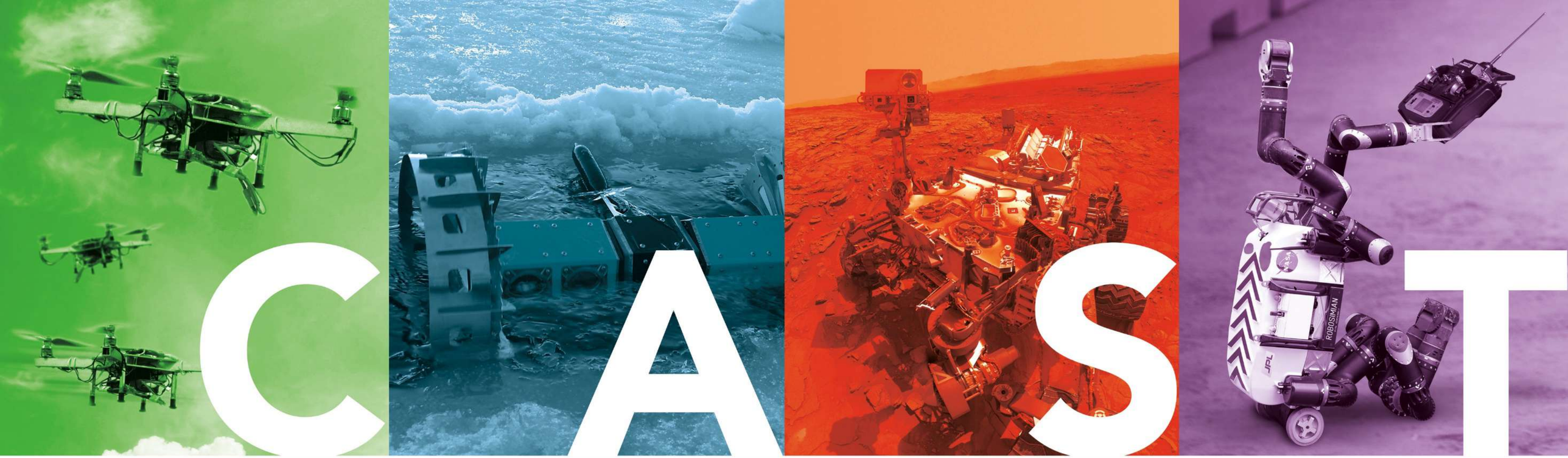
Climate dataset



TENSORLY: HIGH-LEVEL API FOR TENSOR ALGEBRA



- Python programming
- User-friendly API
- Multiple backends: flexible + scalable
- Example notebooks in repository

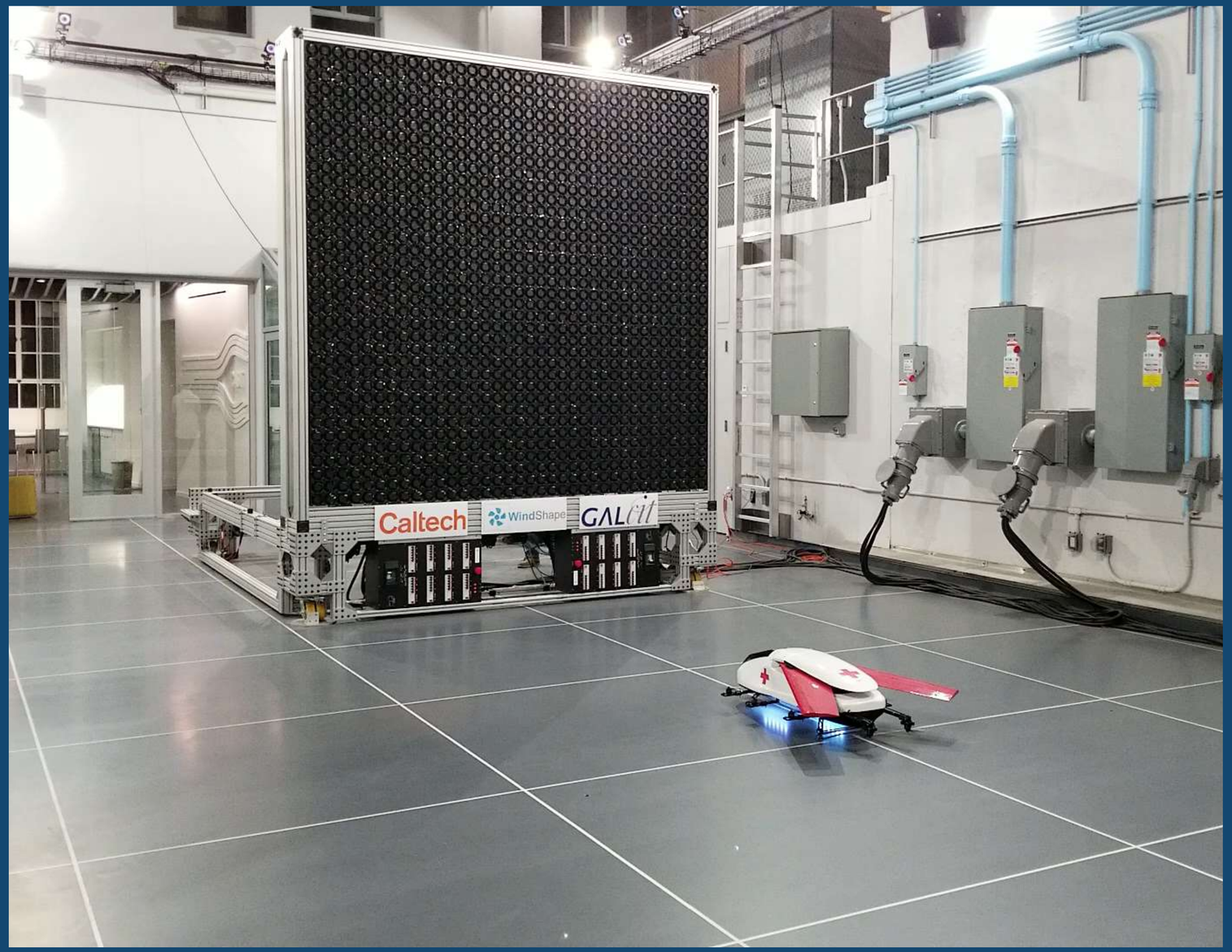


Center for Autonomous Systems and Technologies

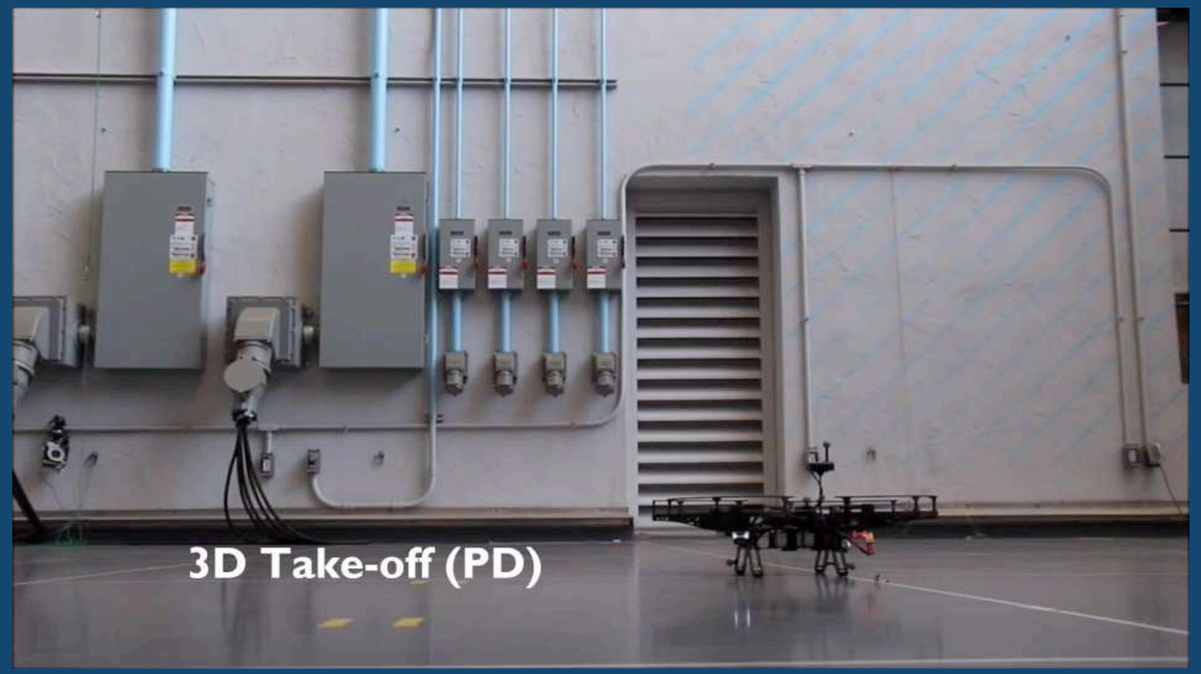
A New Vision for Autonomy

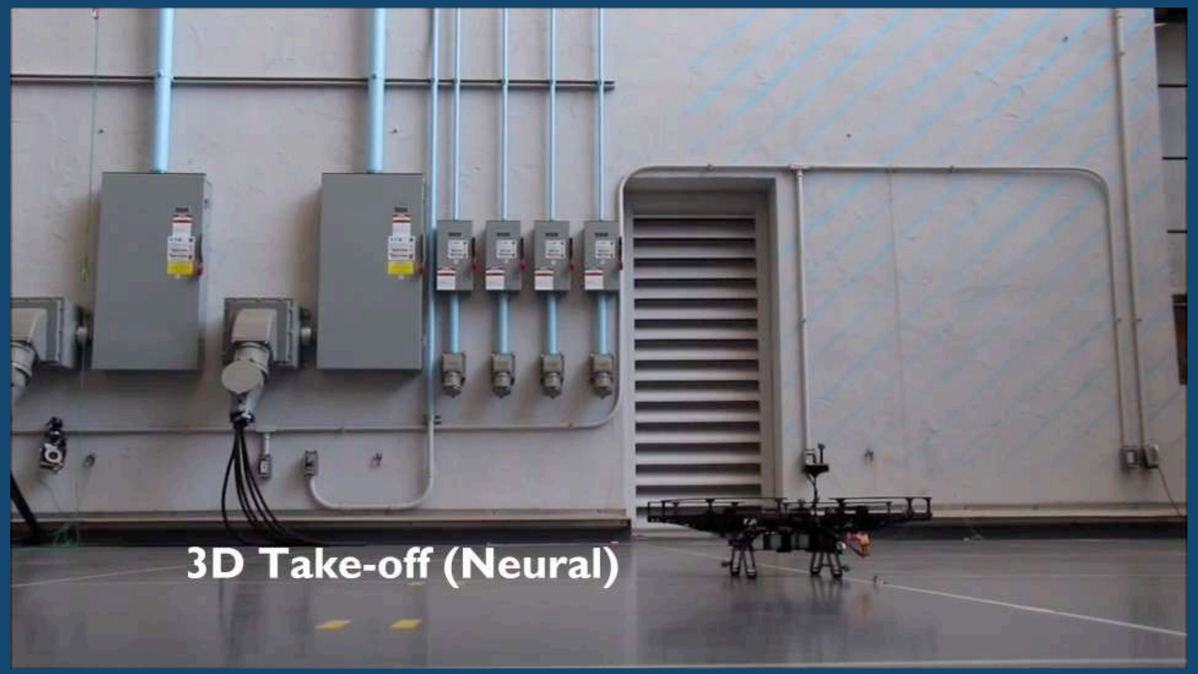
Caltech

CAST: BRINGING ROBOTICS AND AI TOGETHER



FIRST SET OF RESULTS: LEARNING TO LAND






NVIDIA

SOME RESEARCH LEADERS AT NVIDIA

Chief Scientist

Bill Dally

Graphics

Dave Luebke

Alex Keller

Aaron Lefohn

Learning & Perception

Jan Kautz

Robotics

Dieter Fox

Computer vision

Sanja Fidler

Core ML

Me !

Applied research

Bryan Catanzaro

Programming

Michael Garland

Networks

Larry Dennison

Architecture

Steve Keckler

Dave Nellans

Mike O'Connor

VLSI

Brucek Khailany

Circuits

Tom Gray

CONCLUSION

AI needs integration of data, algorithms and infrastructure

- **DATA**
  - **Collection:** Active learning and partial feedback
  - **Aggregation:** Crowdsourcing models
  - **Augmentation:** Graphics rendering + GANs, Symbolic expressions
- **ALGORITHMS**
  - **Convergence:** SignSGD has good rates in theory and practice
  - **Scalability:** SignSGD has same variance reduction as SGD for multi-machine
  - **Multi-dimensionality:** Tensor algebra for neural networks and probabilistic models.
- **INFRASTRUCTURE:**
  - **Frameworks:** Tensorly is high-level API for deep tensorized networks.

COLLABORATORS (LIMITED LIST)

Thank you