Parallel Online Learning

Daniel Hsu Nikos Karampatziakis John Langford

Rutgers University Cornell University Yahoo! Research

Online Learning

- Learner gets the next example x_t, makes a prediction p_t, receives actual label y_t, suffers loss $\ell(p_t, y_t)$, updates itself
- Simple and fast predictions and updates

$$p_t = w^\top x_t$$
$$w_{t+1} = w_t - \eta_t \nabla \ell(p_t, y_t)$$

- Online gradient descent asymptotically attains optimal regret
- Online learning scales well . . .

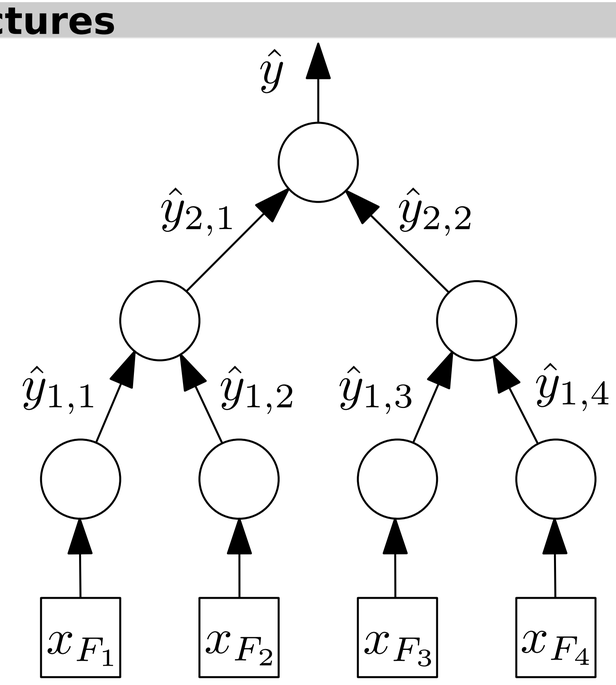
Parallel Online Learning

- . . . but it's a sequential algorithm
- What if examples arrive very fast?
- What if we want to train on huge datasets?
- We investigate ways of distributing predictions, and updates while minimizing communication.

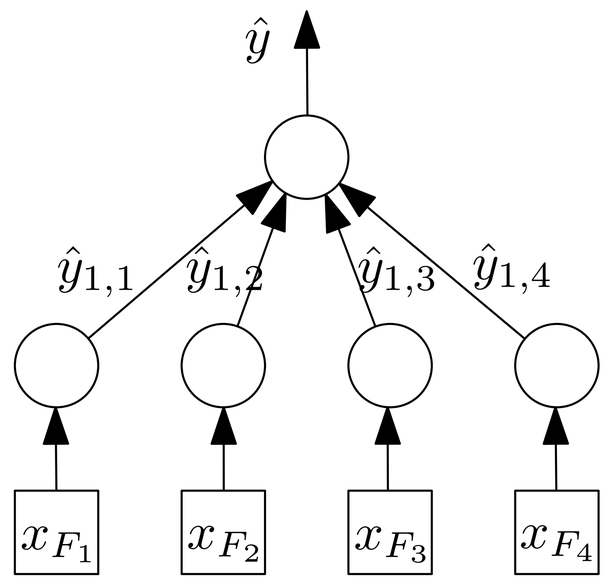
Delay

- Parallelizing online learning leads to delay problems.
- This is exacerbated in a setting with temporally correlated or adversarial examples.
- We investigate no delay and bounded delay schemes.

Tree Architectures



Each node has $f + 1$ weights where f is the node's fan-in. Bottom nodes use subsets of raw features. Others use predictions of their children.



Label travels together with prediction, available in each node

Local Updates

Each node in the tree:

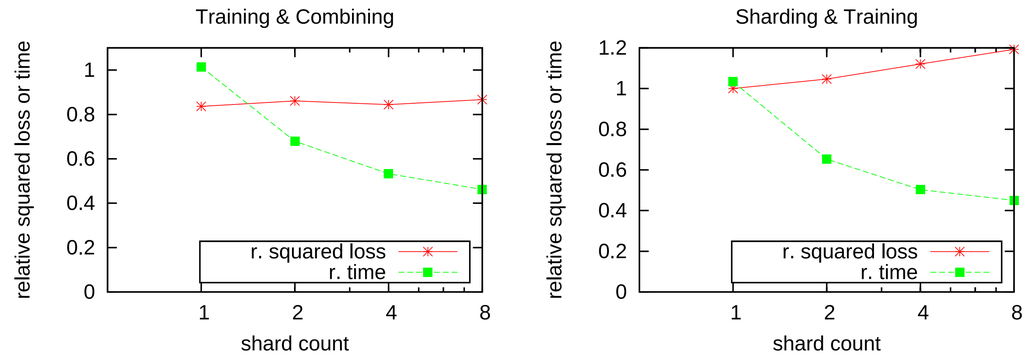
- Computes its prediction $p_{i,j}$ based on its weights and inputs
- Sends $\hat{y}_{i,j} = \sigma(p_{i,j})$ to its parent[a]
- Updates its weights based on $\nabla \ell(p_{i,j}, y)$

No delay

Limited representation power: between Naive Bayes and centralized linear model.

[a] The nonlinearity introduced by σ has an interesting effect

Some Experiments with Local Updates

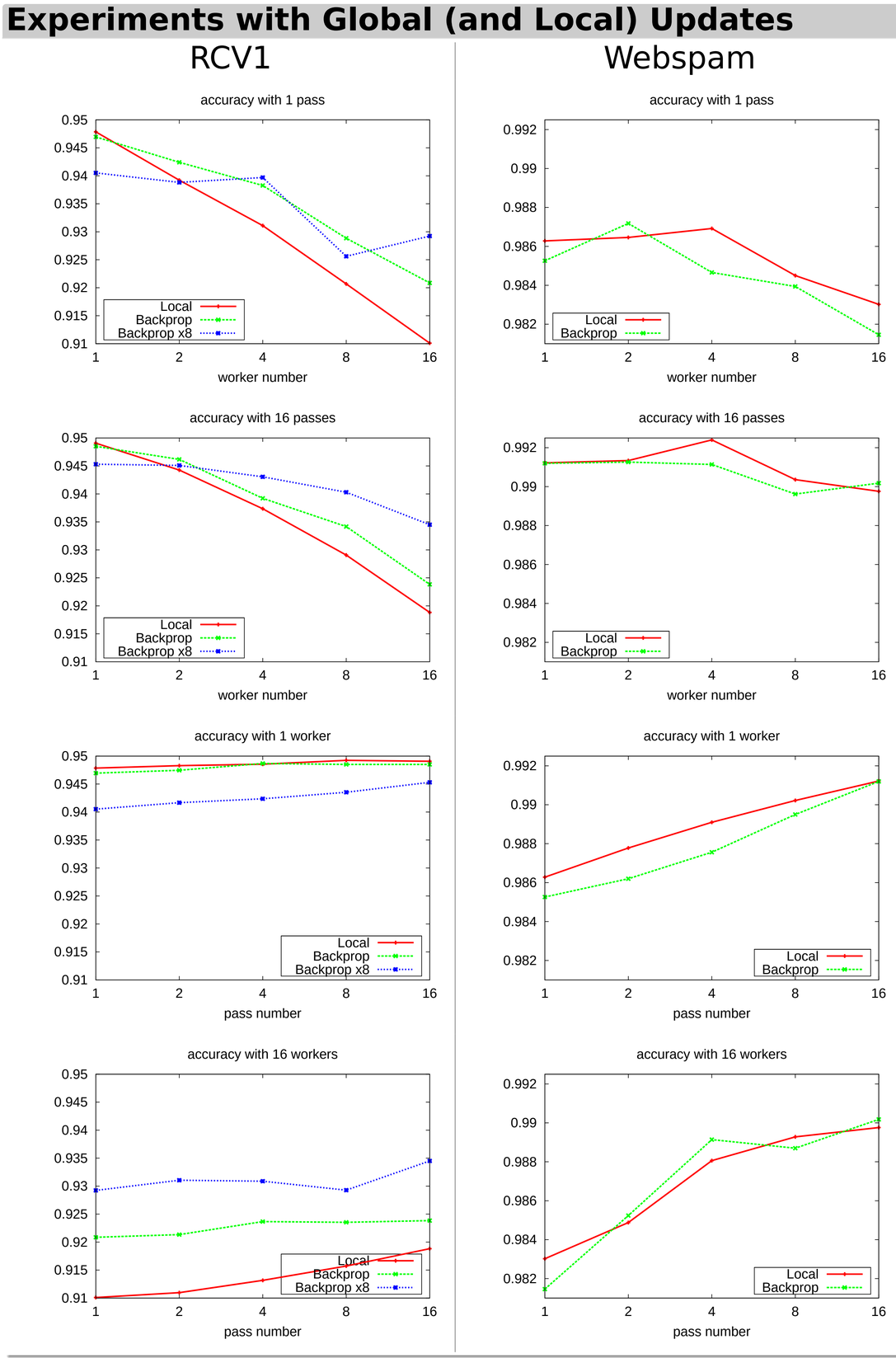


Improvement due to nonlinearity σ

Global Updates

- Unfortunately, local updates can also hurt performance.
- Improved representation power by global training.
- Slightly more communication, some delay.
- Delayed global training
  - Each node predicts but doesn't immediately train on y.
  - Later it receives global prediction $\hat{y}$ and trains as if it predicted that.
- Delayed backprop
  - The tree can be thought as a neural network
  - Lockstep backpropagation would be slow
  - Each node trains locally, sends prediction after training.
  - Later it receives global gradient from parent uses chain rule as in backprop.
- Delay fixed (helps stability, development and debugging)

Experiments with Global (and Local) Updates



Vowpal Wabbit

This (and more) is implemented in Vowpal Wabbit.

http://hunch.net/~vw

Fork it from http://github.com/JohnLangford/vowpal_wabbit

Email: djhsu@rci.rutgers.edu, nk@cs.cornell.edu, jl@yahoo-inc.com WWW: http://hunch.net/~vw