Large-scale Parallel Training in Speech Recognition

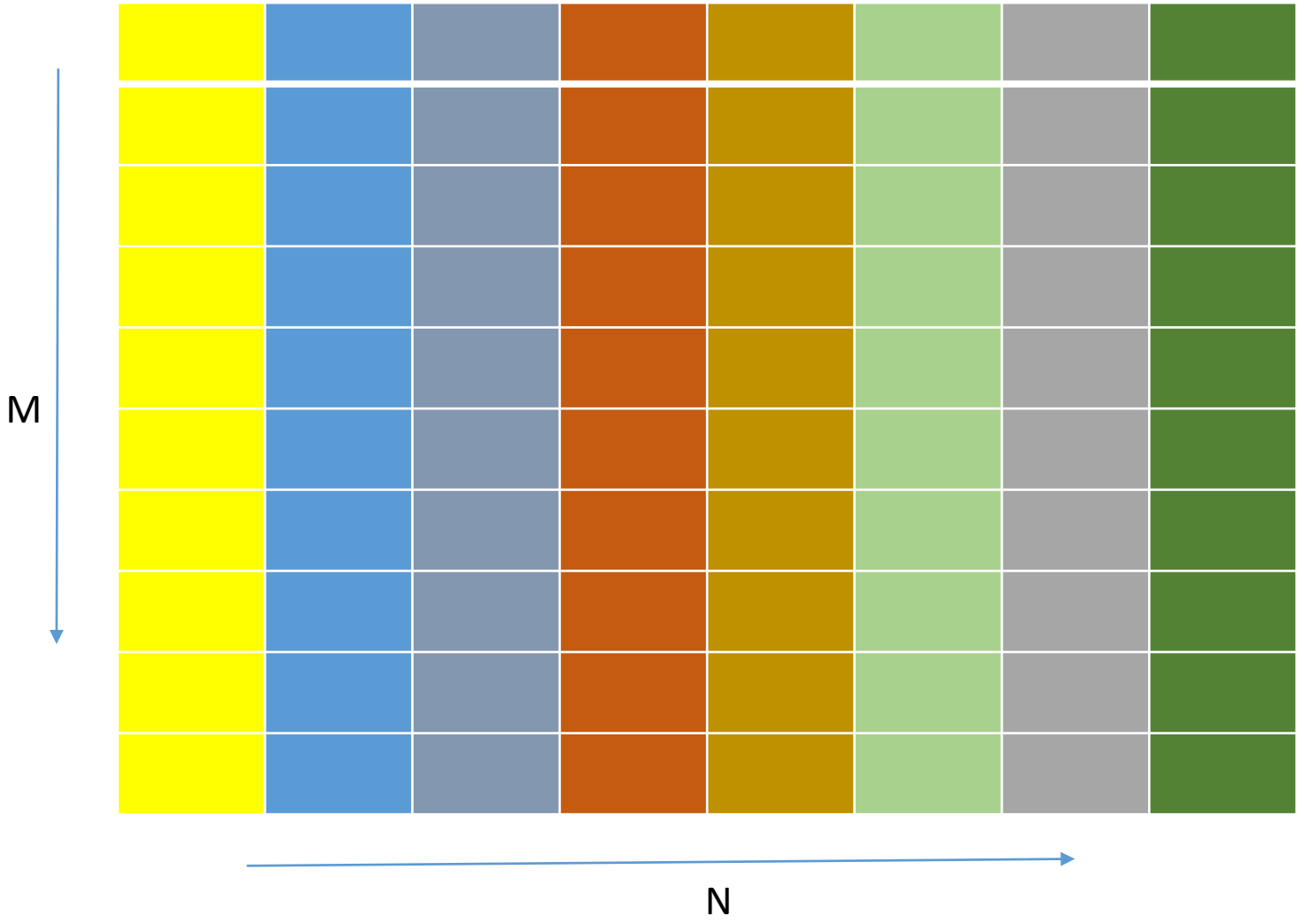
Zhang Zhiyong

2016/01/04

Outline

- Data Preparing
- NN Parallel Training
- Adaptive Gradient Stochastic Optimization
- Sequence Training

Data Preparing



N: Number of parallel jobs

M: outer iterations per epoch

- Randomly distributing data into N by M blocks

NN Parallel Training

- Model parallelism
  - Google DistBelief
- Data parallelism
  - Google DistBelief
  - Baidu DeepSpeech
  - KALDI-nnet3
    - -- Parameter averaging
    - -- Effective learning-rate

NN Parallel Training

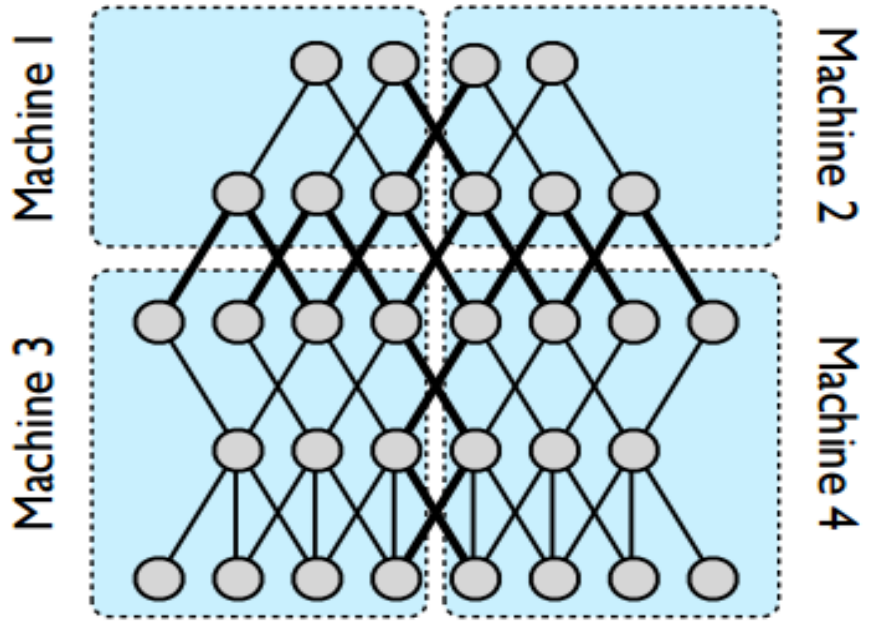


Figure 1: An Example of Model parallelism

NN Parallel Training

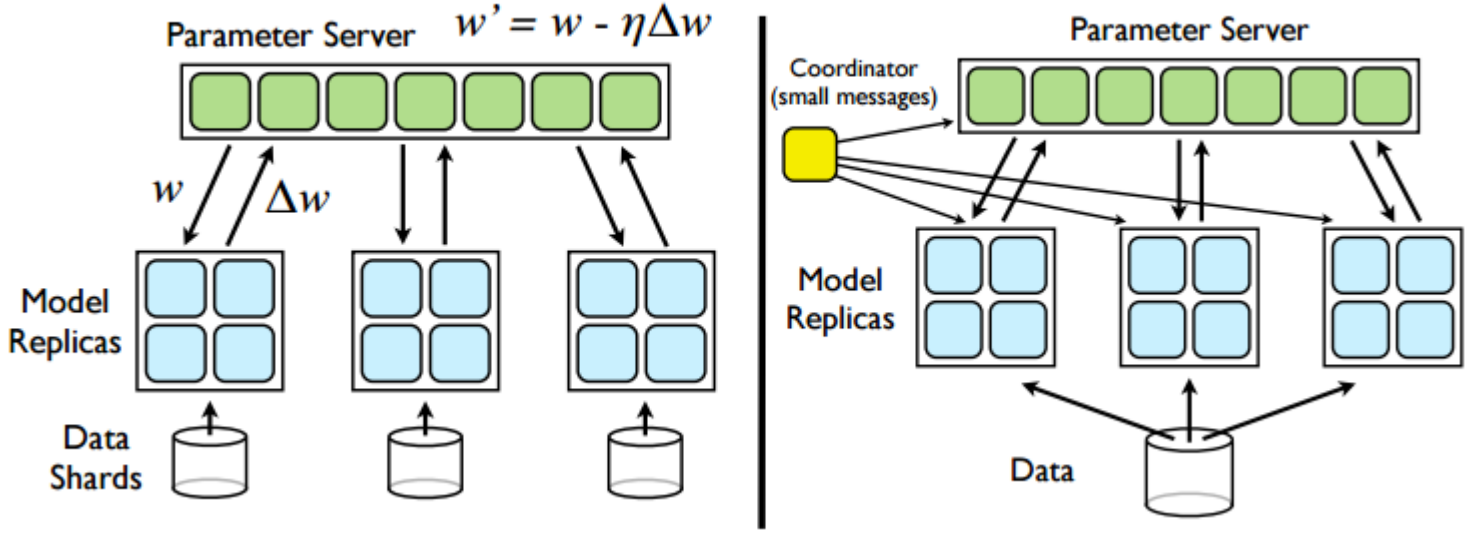


Figure 2: Combination of Model parallelism and Data parallelism

Adaptive Gradient Stochastic Optimization

- Stochastic Gradient Descent(SGD)
- Momentum
- Nestrov
- Adagrad/AdagradMax
- AdaDelta
- AdaM/AdaMax
- Natural Gradient SGD(NG-SGD)

NG-SGD—Metric tensor

- Metric tensor
  - Let $s = \{w \in R^n\}$ be a parameter space on which a function $L(w)$ is defined, the squared length for a small vector dw connecting w and $w + dw$ is:

$$|dw|^2 = \sum_{i,j} g_{ij}(w) dw_i dw_j.$$

  - For Euclidean orthonormal space, the metric tensor is

$$g_{ij}(w) = \delta_{ij} = \begin{cases} 1, & i = j, \\ 0, & i \neq j \end{cases}$$

So the $n \times n$ matrix $G = (g_{ij})$ is the unit matrix I.

NG-SGD—Metric tensor

- For non-orthonormal space—Riemannian space, the metric tensor is called *Riemannnian metric tensor*.

- The steepest descent direction in a Riemannian space

$$-\tilde{\nabla} L(\boldsymbol{w}) = -G^{-1}(\boldsymbol{w}) \nabla L(\boldsymbol{w})$$

where ∇L is the conventional gradient,

$$\nabla L(\boldsymbol{w}) = \left(\frac{\partial}{\partial w_1} L(\boldsymbol{w}), \ldots, \frac{\partial}{\partial w_n} L(\boldsymbol{w}) \right)^T,$$

NG-SGD—Fisher Information matrix

- SGD update formula:

$$\boldsymbol{\theta}_{t+1} = \boldsymbol{\theta}_t + \eta_t \mathbf{g}_t$$

- NG-SGD update formula:

$$\boldsymbol{\theta}_{t+1} = \boldsymbol{\theta}_t + \eta_t \mathbf{E}_t \mathbf{g}_t$$

Where E_t is a symmetric positive definite matrix, optionally inverse Fisher Information Matrix.

Sequence Training

- Maximum likelihood(ML)
- Maximum Mutual Information(MMI)
- Minimum Phone Error(MPE)
- Minimum Word Error(MWE)

Maximum likelihood(ML)

- Maximum Objective function:

$$\mathcal{F}_{\mathrm{MLE}}(\lambda) = \sum_{r=1}^{R} \log p_{\lambda}\left(\mathcal{O}_r | s_r\right)$$

- s_r is the correct sentence
- This is the likelihood of the observations of training data given the correct-transcription.

Maximum Mutual Information(MMI)

- Maximum Objective function:

$$\mathcal{F}_{\mathrm{MMI}}(\lambda) = \sum_{r=1}^{R} \log \frac{p_\lambda\left(\mathcal{O}_r | s_r\right)^\kappa P(s_r)^\kappa}{\sum_s p_\lambda\left(\mathcal{O}_r | s\right)^\kappa P(s)^\kappa}$$

- Equals the posterior probability of the correct sentence s_r.

Minimum Phone Error(MPE)

- Maximum Objective function:

$$\mathcal{F}_{\mathrm{MPE}}(\lambda) = \sum_{r=1}^{R} \frac{\sum_s p_\lambda(\mathcal{O}_r|s)^\kappa P(s)^\kappa A(s, s_r)}{\sum_u p_\lambda(\mathcal{O}_r|u)^\kappa P(u)^\kappa}.$$

- An average of phone accuracy, weighted by sentence likelihood
- $A(s, s_r)$ is the raw phone transcription accuracy of the sentence s given the reference sentence s_r, which equals the number of reference phones minus the number of errors.
- sMBR/MPE

Comparison of Objective functions

Suppose correct sentence is “a”, only alternative is “b”.

Let $a = p_\lambda(o|\mathrm{a})P(\mathrm{a})$(acoustic & LM likelihood), same for “b”.

- ML objective function = $\log(a) +$ other training files
- MMI objective function = $\log\left(\frac{a}{a+b}\right) +$ other training files
- MPE objective function = $\frac{a\times 1+b\times 0}{a+b} +$ other training files

Main Reference

Discriminative training for large vocabulary speech recognition;

Natural gradient works efficiently in learning;

Parallel training of DNNs with natural gradient and parameter averaging;

http://kaldi.sourceforge.net/index.html

Thanks