Parallel Data Mining

Team 2 – Flash Coders
Team Research Investigation
Presentation 1

Foundations of Parallel Computing
16 Sept 2014
Agenda

• Team Members
• Overview
• Computational Problem
• Sequential Algorithm
• Parallel Algorithm
• Reference Papers
Team Members
Overview

Big data analytics - data mining using parallel machine learning algorithms

Application dimensions:

Figure from BIG CPU, BIG DATA: Solving the World's Toughest Computational Problems with Parallel Computing. Alan Kaminsky, 2014.
Why do we need Big Data?

• Data—not the algorithm used—is what matters
• Better to use all the data than to sample it
Where is it?

• Wal*Mart – 20 million point-of-sale transactions a day
• Facebook – 300 petabyte data warehouse that adds 600 terabytes a day
• YouTube – 4 billion views per day
• Instagram – 40 million photos per day

To learn from this data and others, we need to be able to analyze it faster than it arrives in order to make predictions & classifications in real time
Computational Problem

• Linear regression, logistic regression, & neural networks:
  – All try to minimize a cost function by adjusting certain parameters
  – All can be used for regression or classification problems
  – All require summations over the entire dataset which can take a very long time (in batch form)
  – All can get stuck in bad local optima and require several restarts to ensure good results
Gradient descent can take a long time to converge with many training examples, and when it does it may not have found the global optimum.

Sequential Algorithm I

- Linear/logistic regression cost function:
  \[
  \min_{\theta} J(\theta) = \frac{1}{2m} \sum_{i=1}^{m} (h_\theta(x^i) - y^i)^2
  \]

- Algorithm
  - Randomly initialize \( \theta \)
  - Repeat until convergence (for all \( j=0,1,...,n \)):
    \[
    \theta_j = \theta_j - \alpha \frac{1}{m} \sum_{i=1}^{m} (h_\theta(x^i) - y^i)x_j^i
    \]

What if \( m \) equals 500 million training examples!

This term is \( \frac{d}{d\theta_j} J(\theta) \)
Sequential Algorithm II

• Predicted output is then $h_\theta(x^i)$ with trained parameters $\theta$
  
  – Linear Regression:

  $$h_\theta(x^i) = \sum_{j=1}^{n} \theta^i x^i$$

  – Logistic Regression

  $$z = \sum_{j=1}^{n} \theta^i x^i$$
  $$h_\theta(x^i) = \frac{1}{1 + e^{-z}}$$
Parallel Algorithm

• Two ways to parallelize:
  – Within a cluster node: can use multiple cores to do the summation with a parallel reduction before each update to the parameters $\theta$
  – Between cluster nodes: can use a different random initialization on each cluster node and take the best result as final solution
Global Data on Events, Location and Tone (GDELT) dataset contains over a quarter-billion records of global events from 1979 to the present.

- Created with natural language processing of the daily articles of major international news sources.
- Goldstein scale defines the violence and impact of an event on a scale from -10 to +10, which can be used to separate the most violent and destructive global events.
- Recently made SQL accessible through Google BigQuery API.

URL: http://gdeltproject.org/
Visualizing GDELT

Violent worldwide events: Nov 23-30, 2013

Zoomed in view of the Middle East with clickable events
In this paper, the algorithm they present finds a more optimal convergence rate on large scale datasets by distributing the mini-batch (summation over a subset) training of the dataset. Both serial and distributed approaches show improved efficiency of convergence using this technique.
Logistic regression is expensive in terms of both time and storage for large datasets. This paper explores platform and algorithmic optimizations for LR training. It uses distributed platforms like Hadoop and paradigms like MapReduce to parallelize and make LR training on large data sets more efficient.
In this paper, the authors have presented a parallel algorithm based on a MapReduce framework that evaluates the model’s input features. They present a new heuristic and a parallel algorithm for feature evaluation which results in better performance and reduced computational cost.
Further References

  – URL: http://dl.acm.org/citation.cfm?id=1073017

  – URL: http://dl.acm.org/citation.cfm?id=744383

QUESTIONS