Hadoop を用いた分散処理性能の考察

経営工学主専攻　201113031　趙　陽

指導教員：張　勇兵　教授

1. 研究背景と目的

我々の生活の中では至る所に様々なデータが存在する。近年、情報処理技術の発展に伴い、一部のデータはその量が膨大に増え続け、その結果、従来技術によるデータ処理は困難になるほど大きくなっている。本研究ではそれらのデータをビックデータと呼ぶ。その例として、タクシー業界における日々の営業記録がある。その営業記録はテキストデータとして日々蓄積され、その量は膨大なものとなっている。

2004 年に Google は大規模データ、即ち、“ビックデータ”を処理し、データ検索サービスを実現するための基本技術である Google File System 技術を提案した。しかし、Google 社の MapReduce 技術の詳細が公開されていないため、Doug Cutting 氏は Google 社の技術をオープンソースとして実装した。[1]

Hadoop は Mapreduce を実装することによって、複数のコンピュータ機器によって構築されたクラスタ上で、大規模なデータを並列処理することができる。本研究の目的は、まず、複数台のパソコンを使って無料で入手できる分散処理システム Hadoop による大規模なデータを処理する環境を構築することである。また、データの並列処理および処理データ量により、処理時間に対する影響を調べ、Hadoop を利用することでどれだけデータ処理性能を向上できるかをはっきりさせることである。

本研究では、Hadoop の計算時間に影響するデータサイズと計算機台数と仮定する。実際にデータ処理時間はどの程度に変化するのかを考察する。

2. 方法

2.1　実験環境

本研究の実験で用いた Hadoop は 4 台のパソコンからなる。パソコンのオペレーティングシステムは Windows7 であり、その上に Linux 仮想マシン Ubuntu を導入した。Hadoop 環境は Ubuntu 上で構築され、1 台のマシンを Master とし、他の三台のマシンを Slave とした。

表 1　実験環境

物理マシン

OS	Windows 7　professional
メモリ	8GB
プロセッサ	Intel(R)Core(TM)i7 CPU3.40GHz

仮想マシン

仮想マシン	Oracle VM vitualbox -4.3.10-93012
OS	Ubuntu 12.04LTS(64bit)
メモリ	5185MB
プロセッサ	Intel(R)Core(TM)i7 CPU3.40GHz
プロセッサ数	1
Hadoop	Version 1.2.1
Java	Version6
Eclipse	Standard 4.3.2

Master は、NameNode、JobTraker、DataNode、Tasktracker4 つの部分から構成される。Slave は DataNode と TaskTracker2 つの部分から構成される。

クライアントが Master の HDFS に入力ファイルを書き込む。入力ファイルのデータが HDFS のブロックサイズを超えると、ファイル作成リクエストが NameNode に送られる。そのリクエストを受けて NameNode は DataNode にファイル作成を指示し、そのデータを管理する。[2]

Master はデータを Slave の MapReduce に分配し、自分の MapRedecue にも配分する。つまり、Master は Slave と同時にデータ処理を行う。Master の JobTracker がジョブをタスクに分解し、各 TaskTracker に処理を分配する。Master の JobTracker は Map の処理を行う。具体的は、JobTracker は DataNode からデータをもらい、英文から一行ごとに分割する。そして、英文の単語を抽出し、単語を key とする定数とのペアを生成して、Reduce に渡す。各 TaskTracker は Redecue の処理を行う。Redecue の処理では、抽出された単語を key ごとに受け取る回数をカウントする。カウントすることで単語の出現回数を計算する。

2.2　実験用プログラム

Hadoop によるデータ処理の性能を調べるため、WordCount のプログラムを利用する。

WordCount の擬似コード

```
map(key,line){
  i=0, word[] = line を単語毎の分割し、配列に格納；
  while(!(word[i]==null))
```

```
        key = wod[i], i++ ,  output(key, 1);
}
reduce(key, values){
   sum = 0 ;
   for(value に values を要素がなくなるまで代入)
        sum++;
        output(key, sum) ;
}
```

英文のテキストをデータとして受け取り、その単語の出現回数をカウントする。Map では入力データである英文から一行ごとに分割されたものを入力データとして受け取り、その単語を key とする定数とのペアを生成して、Reduce に渡す。Reduce は Map から出されたデータを key ごとに受け取る回数をカウントすることで値を集約する。

3. 結果と考察

本研究では、Hadoop の計算時間に影響するデータサイズおよび計算機台数と仮定する。実際にデータ処理時間はどの程度に変化するのかを考察した。実験の入力テキストデータとして、データサイズが 2MB である“The Brothers Karamazov”Fyodor Dostoyevsky 氏の小説を使った。異なるデータサイズにより、Hadoop の性能に対する影響を調べるため、上記テキストコピーすることにより、200MB、400MB、500MB、800MB、1.6GB、3.2GB のデータファイルを作成した。

実験 1　パソコン台数による影響

実験 1 の条件、入力データを 500MB のファイルと一定にする。計算機の台数は 1 台から 4 台までである。各台数に対し、50 回の実験を行った。

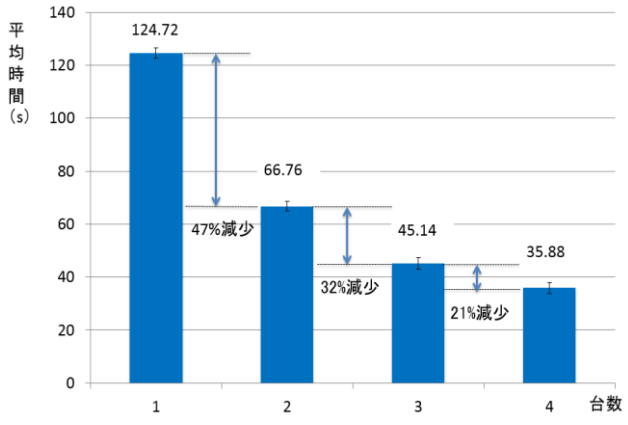

図 1　実験 1 結果

一般には、データサーズが一定の場合は、処理時間は比列に減少することが予想される。例えば、２台は１台の処理時間は 50%減少する。しかし、実際の結果を考察すると、２台の処理時間は１台に比べ、46.47%しか減少しなかったことが分かった。その原因の一つとして、ネットの通信状態である。なぜなら、master は slave と通信するため,データは分割した後、データの送信には時間の損失があった。master は自分もデータ計算処理を行ったため、hadoop 全体の処理時間が伸びた。

実験 2　データサイズによる影響

実験 2 では、計算機の台数は 4 台と固定し、データサーズ 200MB, 400MB, 800MB, 1.6GB, 3.2GB である。各データサイズに対し、50 回の実験を行った。

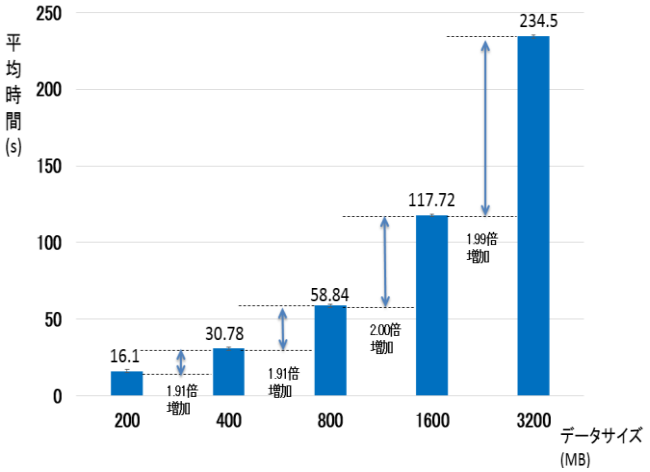

図 2　実験 2 結果

計算機の台数が一定の場合、データサイズの増加に伴い、処理時間は増加する。データサイズによる影響は極めて少ない。

4. まとめ

本研究では、大規模な処理に利用する Hadoop を用いて実験を行い、Hadoop の機能を実際に実現した。Hadoop の処理時間に影響する条件を予想し、収集したデータと比較した。実験の結果から、Hadoop を用いたデータ処理の高速化には、データサイズと計算機の台数が影響することが分かった。具体的には以下のことが分かった。1. データサイズが一定の場合、計算機の台数が増えるに従って、処理時間は短縮するが、その割合が小さくなっていく。2. 計算機の台数が一定の場合、データサイズの増加に伴い、処理時間は増加するが、データサイズによる影響は極めて少ない。

今後の課題

本研究では、現実に存在するビックデータを用い、それを処理するために現在使用されているアルゴリズムを実装していない。そのため、Hadoop の性能を十分に検証できたとは言えない。現実に即したビックデータを処理することによって、Hadooop の性能をより具体的かつ正確に検証することを今後の課題とする。

参考文献

[1] Tom Wbite 著. 玉川竜司 兼田聖士 訳. Hadoop 第二版, オライリージャパン, 2012

[2] 佐々木達也 . Hadoop ファーストガイド, 秀和システム社. 2012