Big Data-Programmering

Sammanfattning

”Big Data”; datamängder av sådan volym att konventionella metoder blir otympliga ställer andra krav för hantering än mindre datamängder. En liten fördröjning i hanteringen multipliceras miljontals gånger över traverseringen av en datamängd. För analys av dessa datamängder krävs tidseffektiva och effektiva metoder för filtrering och lagring. Med konceptet MapReduce kan man åtskilligt reducera antalet operationer genom att sortera information för ens ändamål och sedan använda den mindre mängden för att lösa problemet. Lagring av data måste även ske på ett så pass bra sätt att du snabbt kan komma åt information samt utöka storleken i takt med att datamängden ökar. Big Data ställer även vissa krav på programspråken men inte bara på hastighet utan även på tillgänglighet. Dataanalys har stora fördelar både för företag och för vetenskap. För att tillmötesgå den växande marknaden krävs programmeringsspråk som är lätta att använda för personer som inte nödvändigtvis jobbar främst med statistik

Innehållsförteckning

1. Inledning

1.1 Bakgrund

Big data handlar i stora drag om lagring, behandling och bearbetning av så pass stora volymer data att användning av traditionell statistisk mjukvara blir otympligt.[1] Några bakomliggande orsaker till dessa nya större datamängder är den ökade användningen av mobila sensorer och mjukvaruloggar.[2] Svårigheten att behandla data har lett till en framväxt av nya angreppssätt där bland annat två av dem är ”MapReduce” och användning av ”NoSQL-databaser”.

1.2 Frågeställning

Är den enda skillnaden mellan hantering av ”Big Data” och vanliga datamängder mängden data?

Krävs nya verktyg för hantering av större datamängder?

Ur programmeringspråksynpunkt:

Går det bra att använda traditionella språk rakt av?

Vilka egenskaper hos språket är viktigast för bäst hantering av datan?

Spelar användarvänlighet någon roll, dvs vem har användning för ”Big data”?

2 Huvuddel

2.1 Big Data.

2.1.1 Definition av Big Data.

”Big data” är intuitivt när man har så pass stora datamängder att det inte går att hantera med vanliga databassystem. Hur stor den här datamängden är varierar och det man kallar för Big data idag kommer förmodligen, i takt med att möjligheten att lagra och hantera data utvecklas, inte kallas det i framtiden. Vad som räknas som Big data beror också på vem eller vilka som ska hantera den, ett mindre företag, med mindre datorkraft, har större svårigheter med datamängder som större företag skulle klara av.[3]

Inom det som man kallar Big data existerar även stor variation av vad den faktiska datan består i. Den kan till exempel vara i form av ljud eller bild och inte nödvändigtvis bara i form av siffror.[4]

En annan egenskap som kännetecknar Big data är behovet att snabbt kunna komma åt information även vid stora datamängder[5], ett exempel är aktiebörser där aktiepriserna behöver kunna uppdateras på några millisekunder för alla användare.[6]

[1] http://www.ijis.net/ijis7_1/ijis7_1_editorial.pdf
[2] http://en.wikipedia.org/wiki/Big_data
[3] http://en.wikipedia.org/wiki/Big_data
[4] http://www.mongodb.com/big-data-explained
[5] http://en.wikipedia.org/wiki/Big_data
[6] http://www.mongodb.com/big-data-explained

2.1.2 Exempel med Big Data.

Ett exempel där enorma mängder data samlas in och sållas är ”Large Hadron Collider Experiment”. Där används sensorer för att lagra information om kollisioner mellan partiklar. Nära 600 miljoner kollisioner per sekund sållas ned till endast 100 relevanta kollisioner per sekund.[7] Här används alltså mindre en 0,001% av informationen från sensorerna. Här ser man hur viktigt det är med hantering och bearbetning av datan.

I och med att fler sensorer börjar användas så växer även datamängderna. En Boeing 737 kan till exempel generera 240 terrabyte data under en enda flygning.[8] Även användandet av smartphones och sociala medier har lett till ökade datamängder. Facebook har till exempel hand om över 50 miljarder foton från användare.[9]

2.1.3 Hantering av Big Data.

När det gäller hantering av Big data är lagring, bearbetning och åtkomst de viktigaste delarna. Man vill kunna lagra datan effektivt, sålla bort onödig information och snabbt kunna ta fram relevant information.

För att kunna lagra stora datamängder krävs att man kan få tillgång till mer uttrymme i takt med tillväxten av data. Sättet man lagrar det på måste också göra det möjligt att komma åt datan tillräckligt snabbt, detta för att kunna köra analytiska verktyg.[10] Större företag använder sig av så kallade ”Hyperscale computing environments”, dessa miljöer använder sig ofta av analytiska motorer som till exempel ”NoSQL”.

NoSQL är en typ av databas som används för hantering av större mängder data. NoSQL skiljer från vanliga relationsdatabaser där tabeller refererar till andra tabeller med hjälp av nycklar. I vissa typer av NoSQLdatabaser kan till exempel samla ihop data från flera tabeller in till en enda fil.[11]

Den stora fördelen med NoSQLdatabaser är möjligheten att enkelt skala upp i och med att ens mängd data växer. Man behöver bara lägga till fler databasservrar för att öka ens kapacitet. NoSQLdatabasen ses som endast en databas oavsett om du lägger till fler servrar.[12]

Ett annat verktyg för hantering av stora datamängder är ”MapReduce”. MapReduce är en model för programmering som genom att använda sig av en Map funktion för filtrering och sortering, samt en reduce funktion som summerar.[13]

Mapfunktionens uppgift i MapReduce är att från ett givet nyckel/värde par kunna skapa en ny mängd med ny nyckel/värde sorterad för vad man vill uppnå. ReduceFunktionen ska sedan samla ihop alla värden med den nya nyckeln.[14]

I ”MapReduce: Simplified Data Processing on Large Clusters”[15] ger författarna följande kodexempel:

---

[7] http://en.wikipedia.org/wiki/Big_data
[8] http://www.mongodb.com/big-data-explained
[9] http://en.wikipedia.org/wiki/Big_data
[10] http://www.computerweekly.com/podcast/Big-data-storage-Defining-big-data-and-the-type-of-storage-it-needs
[11] http://www.couchbase.com/nosql-resources/what-is-no-sql
[12] http://www.couchbase.com/nosql-resources/what-is-no-sql
[13] http://en.wikipedia.org/wiki/MapReduce
[14] https://www.usenix.org/legacy/publications/library/proceedings/osdi04/tech/full_papers/dean/dean_html

```
map(String key, String value):
    // key: document name
    // value: document contents
    for each word w in value:
      EmitIntermediate(w, "1");

reduce(String key, Iterator values):
    // key: a word
    // values: a list of counts
    int result = 0;
    for each v in values:
      result += ParseInt(v);
    Emit(AsString(result));
```

Pseudokoden löser problem att, från en stor samling dokument, räkna hur många gånger ett ord finns med. Mapfunktionen samlar in alla hittade ord och Reduce funktionen räknar ihop resultatet.

Fördelen med MapReduce är som med NoSQL att den är skalbar vilket är en viktig egenskap när man behandlar stora datamängder.

2.2 Big Data och programmeringsspråk.

2.2.1 Big Data och programspråket R.

Programspråket R är ett programsspråk främst använt för statistik och utvinning av data.[16] Till Programspråket finns även en serie med paket kallade ”Programming with big data in R” (pbdR) som är specielt anpassad för användning inom ”Big data”- sammanhang. PbdR skiljer sig från R på det sättet att pbdR inriktar sig på analys av data som distribuerats så att flera datorer kan arbeta med samma analys samtidigt.[17] Detta kallas för ”Massively Parallel Computing”[18] och fungerar genom att använda sig av ett så kallat ”Message Passing Interface” (MPI)[19] som datorerna i det parallella nätet kommunicerar via.

Parallel Computing är alltså ett sätt att få flera datorer att arbeta på samma problem samtidigt.[20] Detta bygger på att stora problem ofta går att dela upp i flera minder bitar. Datorer kan då samarbeta och lösa de mindre delproblemen parallellt med varandra.

Implementation av detta i PbdR existerar på huvudsakligen två sätt; Rmpi och pbdMPI.[21]

I Rmpi-systemet är en processor i nätet den en slags arbetsledare som kontrollerar alla andra processorer. Denna kommunikationsmodel kallas för ”Master/slave”.

PbdMPI-systemet använder sig av ”SPMD”-teknik (singel program, multiple data) där är alla processorer är jämnlika.[22]

---

[15] https://www.usenix.org/legacy/publications/library/proceedings/osdi04/tech/full_papers/dean/dean_html/
[16] http://en.wikipedia.org/wiki/R_(programming_language)
[17] http://en.wikipedia.org/wiki/Programming_with_Big_Data_in_R
[18] http://en.wikipedia.org/wiki/Massively_parallel_(computing)
[19] http://en.wikipedia.org/wiki/Message_Passing_Interface
[20] http://en.wikipedia.org/wiki/Parallel_computing
[21] http://en.wikipedia.org/wiki/Programming_with_Big_Data_in_R
[22] http://en.wikipedia.org/wiki/SPMD

2.2.2 Big Data och programmeringsspråket Python.

Python är ett generellt, användarvänligt programmeringsspråk som stödjer flera programmeringsparadigm.[23]

Användarvänligheten gör det möjligt för fler personer att använda sig av dataanalys då det inte alls kräver lika mycket förkunskaper som R.[24] Detta verkar ha större betydelse för användaren en språkets snabbhet.[25]

Python är ett ”Interpreterande” och ”dynamiskt typat” språk vilket är två egenskaper som gör det långsammare jämfört med andra programspråk som använder sig av en ”kompilerande miljö” och är ”statiskt typade”.[26] Med stora datamängder blir resultatet av en liten fördröjning snabbt väldigt stor när den ska multipliceras med till exempel 600 miljoner varje sekund som i fallet med ”Large Hydron Collider Experiment”.

Det finns dock vissa bitar i R för dataanalys som Python helt saknar. Detta är nödvändigtvis inte något problem i framtiden hävdar vissa[27] då Python är i stor utveckling och nya paket publiceras.

Python tillhandahåller även inbyggda map och reduce funktioner som lätt och effektivt kan kombineras för att lösa problem. I följande programexempel löser artikelskribenten[28] problemet, att räkna alla element ett antal listor, genom de inbyggda map och reduce funktionerna:

```
a = [1,2,3]
b = [4,5,6,7]
c = [8,9,1,2,3]
L = map(lambda x:len(x),[a,b,c])
N = reduce(lambda x,y: x+y,L)
```

Map och reduce delen I ovanstående kod kan till och med skrivas på en enda rad:

```
N = reduce(lambda x,y: x+y, map(lambda x:len(x),[a,b,c]))
```

2.3 Applicering av Big Data.

Det finns många fördelar med Big data, både för företag och institutioner.

2.3.1 Big data och företag.

Kunder lämnar ifrån sig enorma mängder information när de använder till exempel sociala medier. Genom att behandla denna information kan man hitta trender och få statistik som man sedan kan omvandla till vinst genom till exempel riktad reklam. För att göra detta måste man lyckas sortera datan på något sätt. Om man till exempel sorterar vilka produkter som en person köpt efter region får man ut statistik över var vissa produkter är populära.

[23] http://en.wikipedia.org/wiki/Python_(programming_language)
[24] http://insidebigdata.com/2013/12/09/data-science-wars-python-vs-r/
[25] http://www.infoq.com/news/2014/01/bigdata-languages?utm_campaign=infoq_content&utm_source=infoq&utm_medium=feed&utm_term=Programming-news
[26] https://jakevdp.github.io/blog/2014/05/09/why-python-is-slow/
[27] http://readwrite.com/2013/11/25/python-displacing-r-as-the-programming-language-for-data-science
[28] http://mikecvet.wordpress.com/2010/07/02/parallel-mapreduce-in-python/

Den Amerikanska kedjan Target använde sig tidigare av ett system där de tittade på köpvanor hos vissa kunder och skickade sedan hem personliga rabattkuponger till kunderna. De använde informationen för att hitta mönster bland kunder och lyckades hitta ett samband mellan köpta produkter och graviditet. Target använde sig av informationen för att kunna skicka ut rabattkuponger på babysaker innan barnet ens var fött.[29]

Genom att snabbt kunna sortera och klumpa ihop stora mängder data kan man hitta samband och statistik som man sedan kan omvandla till vinst i form av till exempel riktad reklam eller marknadsstatistik[30]

Företaget Google har sedan 2008 en webservice kallad ”Google Flu Trends”[31] där de med hjälp av sökdata gör en förutsägelse av hur många som kommer att bli sjuka.[32] I början av tjänsten så var den jämfört med faktiska siffror till 97% rätt[33] men mellan år 2012 och 2013 förutsåg de dubbelt så många doktorsbesök som den faktiska siffran låg på.[34]

2.3.2 Big Data och Institutioner.

Big data går även att applicera inom forskning och vetenskap. Med hjälp av sensorer kan man samla in till exempel stora mängder meterologisk data. Denna datamängd måste sedan sorteras och sållas för att få fram relevant väderinformation. ”NASA center for climate simulation” har till exempel över 32 petabyte (10^{15}) med klimatobservationer och simuleringar.[35]

2.4 Slutsatser

Är den enda skillnaden mellan hantering av ”Big Data” och vanliga datamängder mängden data?

Jag håller inte alls med ” Att hantera mycket data är samma sak som att hantera lite data; det finns bara mer av det” Min främsta anledning är att man inte kan komma undan med att skriva ineffektiv kod när man arbetar i väldigt stor skala. För att illustera detta använder jag följande exempel:

Om all data från ”NASA center for climate simulation” (32 * 10^{15} byte) skulle innehålla textsträngar på 10 byte styck (32 * 10^{14} strängar) skulle en fördröjning på en microsekund (1 * 10^{-6} s) per textsträng vid en igenomsökning av alla strängar resultera i en total fördröjning av lite över 100 år. Även om exemplet är en aning överdrivet så hjälper det att illustrera min poäng.

En annan poäng är att det man kallar för Big data ofta är data i annat format än textsträngar och siffror. Det är svårt att anpassa traditionella databasmetoder att fungera på större mängder av till

[29] http://www.forbes.com/sites/kashmirhill/2012/02/16/how-target-figured-out-a-teen-girl-was-pregnant-before-her-father-did/
[30] http://theglobalobservatory.org/2013/05/as-private-sector-embraces-big-data-public-sector-falls-behind/
[31] http://en.wikipedia.org/wiki/Google_Flu_Trends
[32] http://www.google.org/flutrends/about/how.html
[33] http://static.googleusercontent.com/external_content/untrusted_dlcp/research.google.com/en/us/archive/papers/detecting-influenza-epidemics.pdf
[34] http://en.wikipedia.org/wiki/Google_Flu_Trends
[35] http://www.csc.com/cscworld/publications/81769/81773-supercomputing_the_climate_nasa_s_big_data_mission

exempel bilder. I fallet med Facebooks över 50 miljarder foton så är inte bara användare kopplade till bilderna utan de använder även algoritmer för igenkänning av ansikten som körs på varje bild.[36]

Exemplet med ”Large Hadron Collider Experiment”, där nära 600 miljoner kollisioner per sekund sållades ned till 100 per sekund, visar också hur viktigt möjligheten att kunna filtrera och sortera datamängder. Vikten av att koncept så som ”MapReduce” åskådliggörs av den här typen av problem. Att snabbt kunna minska datamängden man behöver analysera sparar mycket tid i och med att onödiga operationer helt enkelt hoppas över.

Även om man filtrerar och sorterar sin datamängd så måste man ändå kunna lagra den effektivt så man inte behöver slänga bort viktigt information. Även om viss information vid första anblick kan tyckas irrelevant kan den i kombination av annan statistik bygga kraftfulla modeller som i fallet med ”Google Flu Trends” där med hjälp av bland annat statistiken över hur många som sökt på ordet ”flu” kunde skapa en modell som under en tid ganska bra speglade verkligheten.[37]

För att lagra stora datamängder krävs verktyg så att man någorlunda snabbt kan få åtkomst till information. Annars funkar inte koncept som ”MapReduce”. Till hjälp har man ” Hyperscale computing environments”, miljöer som har den stora fördelen att man kan lägga till mer datorkraft för att öka ens lagringsutrymme i och med att till exempel ens användarbas växer. Det är även viktigt att lagringssättet är ändamålsenligt för att strömlinjeforma det för just de funktioner som man själv behöver.

Ur programmeringsynpunkt har jag främst tittat på två ledande språk inom Big Data-Programmering, Python och Programspråket R med paketet ”Programming with big data in R” (pbdR). Jag skulle sammanfatta det som att anledningen till att personer använder språken är olika. Slutsatsen jag drar är att för tillfället så är R är ett programspråk som är svårt för nybörjare inom programmering att lära sig men som innehåller kraftiga verktyg för dataanalys. Å andra sidan så är Python ett lättare språk att lära sig så det bryter ner barriären och gör dataanalys mer tillgängligt. Personer med annan specialitet kan då utnyttja de stora fördelarna som finns, ekonomiska eller vetenskapliga. Ett problem med Python är dock att det är långsamt men det är till viss del också det som gör det lättillgängligt med exempelvis dynamisk typning. Slutsatsen är alltså att vilket programmeringspråk man bör välja beror helt på ens förutsättningar, vad man vill analysera och hur mycket data man har. De viktigaste faktorerna för ett språk för dataanalys är enligt mig alltså operationshastighet och tillgänglighet.

[36] http://www.extremetech.com/extreme/178777-facebooks-facial-recognition-software-is-now-as-accurate-as-the-human-brain-but-what-now
[37] http://www.google.org/flutrends/about/how.html

2.6 Referencer

"Big Data": Big Gaps of Knowledge in the Field of Internet Science [www] http://www.ijis.net/ijis7_1/ijis7_1_editorial.pdf Hämtat 2014-11-11

Wikipedia: Big Data [www] http://en.wikipedia.org/wiki/Big_data Hämtat 2014-11-11

mongoDB: Big Data Explained [www] http://www.mongodb.com/big-data-explained Hämtat 2014-11-11

ComputerWeekly: Big Data Storage: Defining Big Data and the type of storage it needs. [www] http://www.computerweekly.com/podcast/Big-data-storage-Defining-big-data-and-the-type-of-storage-it-needs Hämtat 2014-11-11

Couchbase: Why NoSQL? [www] http://www.couchbase.com/nosql-resources/what-is-no-sql Hämtat 2014-11-11

Wikipedia: MapReduce [www] http://en.wikipedia.org/wiki/MapReduce Hämtat 2014-11-11

MapReduce: Dean, Jeffrey & Ghemawat, Sanjay (2004): Simplified Data Processing on Large Clusters[www] https://www.usenix.org/legacy/publications/library/proceedings/osdi04/tech/full_papers/dean/dean_html/ Hämtat 2014-11-11

Wikipedia: R (programming language) [www] http://en.wikipedia.org/wiki/R_(programming_language) Hämtat 2014-11-11

Wikipedia: Programming with Big Data in R[www] http://en.wikipedia.org/wiki/Programming_with_Big_Data_in_R Hämtat 2014-11-11

Wikipedia: Massively Paralell(computing) [www] http://en.wikipedia.org/wiki/Massively_parallel_(computing) Hämtat 2014-11-11

Wikipedia: Message Parsing Interface [www] http://en.wikipedia.org/wiki/Message_Passing_Interface Hämtat 2014-11-11

Wikipedia: Parallel Computing [www]

http://en.wikipedia.org/wiki/Parallel_computing Hämtat 2014-11-11

Wikipedia: SMPD [www] http://en.wikipedia.org/wiki/SPMD Hämtat 2014-11-11

Wikipedia: Python (programming language) [www] http://en.wikipedia.org/wiki/Python_(programming_language) Hämtat 2014-11-11

Guitierrez, Daniel (2013): Data Science Wars: Python vs R [www] http://insidebigdata.com/2013/12/09/data-science-wars-python-vs-r/ Hämtat 2014-11-11

Menguy, Charles (2014): Big Data: Do Languages Really Matter? [www] http://www.infoq.com/news/2014/01/bigdata-languages?utm_campaign=infoq_content&utm_source=infoq&utm_medium=feed&utm_term=Programming-news Hämtat 2014-11-11

Vanderplas, Jake (2014): Why Python is Slow: Looking Under the Hood [www] https://jakevdp.github.io/blog/2014/05/09/why-python-is-slow/ Hämtat 2014-11-11

Asay, Matt (2013) : Python Displacing R As The Programming Language For Data Science [www] http://readwrite.com/2013/11/25/python-displacing-r-as-the-programming-language-for-data-science Hämtat 2014-11-11

Cvet, Michael (2010): Parallel MapReduce in Python In Ten Minutes [www] http://mikecvet.wordpress.com/2010/07/02/parallel-mapreduce-in-python/ Hämtat 2014-11-11

Forbes: How Target Figured Out A Teen Girl Was Pregnant Before Her Father Did [www] http://www.forbes.com/sites/kashmirhill/2012/02/16/how-target-figured-out-a-teen-girl-was-pregnant-before-her-father-did/ Hämtat 2014-11-11

O'Reilly, Marie (2013) : As Private Sector Embraces Big Data, Public Sector Falls Behind [www] http://theglobalobservatory.org/2013/05/as-private-sector-embraces-big-data-public-sector-falls-behind/ Hämtat 2014-11-11

Wikipedia: Google Flu Trends [www] http://en.wikipedia.org/wiki/Google_Flu_Trends Hämtat 2014-11-11

Google: Flu Trends: [www] http://www.google.org/flutrends/about/how.html Hämtat 2014-11-11

Google: Detecting Influenza Epedemics using search engine query data. [www] http://static.googleusercontent.com/external_content/untrusted_dlcp/research.google.com/en/us/archive/papers/detecting-influenza-epidemics.pdf Hämtat 2014-11-11

CSC: Climate Change Simulation: Nasa's Weather Supercomputer [www] http://www.csc.com/cscworld/publications/81769/81773-supercomputing_the_climate_nasa_s_big_data_mission Hämtat 2014-11-11