빅데이터 처리 시스템 개관

한기용

목차

- 빅데이터의 정의
- 빅데이터의 예
- 빅데이터 처리방식의 변천
- 빅데이터 시스템의 구성
- 성공스토리들
- 문제점
- 관련회사들

Is Big Data a marketing buzzword?

빅데이터의 정의

정의 1

- "서버 한대로 처리할 수 없는 규모의 데이터"

- 2012년 4월 아마존 클라우드 컨퍼런스에서 아마존의 data scientist인 존 라우저가 내린 정의
  - 분산 환경이 필요하느냐에 포커스

정의 2

- "기존의 소프트웨어로는 처리할 수 없는 규모의 데이터"

- 대표적인 기존 소프트웨어
  - 오라클이나 **MySQL**과 같은 관계형 데이터베이스.
  - 많은 경우 분산환경을 염두에 두지 않음. **Scale-up** 접근방식 **(vs. Scale-out)**

정의 3

- 4V (Volume, Velocity, Variety, Veracity)
  - **Volume:** 데이터의 크기가 대용량?
  - **Velocity:** 데이터의 처리 속도가 중요?
  - **Variety:** 구조화/비구조화 데이터 둘다?
  - **Veracity:** 데이터의 불확실성

- IDC와 같은 컨설팅업체가 가장 많이 사용하는 정의

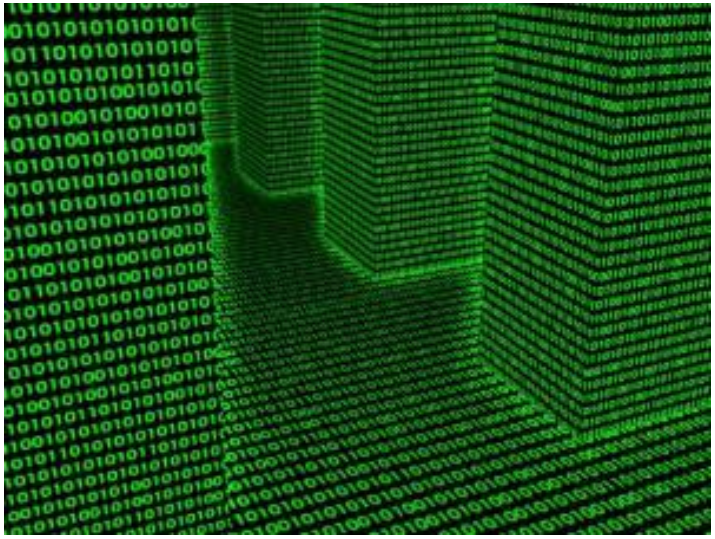
빅데이터의 예

검색엔진 데이터

- 수백억개의 웹페이지 크롤, 인덱싱
- 웹페이지 그래프를 기반으로 페이지랭크 계산
- 사용자 검색어와 클릭로그를 기반으로 한 각종 마이닝
  - 동의어 찾기
  - 통계기반 번역 (statistical translation)
  - 검색입력 자동 완성(auto-completion)

디바이스 데이터

- 모바일 디바이스
  - 위치정보
- 스마트 **TV**
- 각종 센서 데이터
- 네트워킹 디바이스
- …



야후 웹 검색팀에서의 경험한

빅데이터 처리방식의 변천

야후 검색팀의 예 (하둡 사용전)

- 크롤,색인,그래프 생성을 위해서 조금씩 다른 자체개발 소프트웨어들을 사용.
  - 중복투자 및 유지보수의 문제.
    - 세가지 모두 일종의 분산처리시스템으로 자기가 하는 일에 최적화 되었지만 많은 부분에 공통점들이 존재.
  - 야후밖에서 전혀 쓸모가 없음
    - 개인의 스킬셋 제약 및 **Hiring** 관점에서도 문제.
- 검색로그의 경우 용량 문제로 데이터의 전수조사 불가.
  - 마이닝시 샘플링에 의존.
  - 데이터 액세스 자체가 쉽지 않았음.
    - 이것 역시 복잡한 승인 프로세스로 시간이 걸렸음.

하둡이란?

- Doug Cutting이 구글랩에서 발표한 두개의 논문에 기반해 2005년 만든 오픈소스 프로젝트
  - 2003년 The Google File System.
  - 2004년 MapReduce: Simplified Data Processing on Large Cluster.
- 처음 시작은 Nutch라는 오픈소스 검색엔진의 하부 프로젝트.
  - 하둡은 Doug Cutting의 아들의 코끼리 인형의 이름.
  - 2006년에 아파치 톱레벨 별개 프로젝트로 떨어져나옴.
- 크게 분산파일시스템인 HDFS와 분산처리시스템인 MapReduce 두개의 컴포넌트로 구성됨.

야후 검색팀의 예 (하둡도입초반)

- 2006년초 Doug Cutting을 영입하여 하둡 도입을 실험. 20노드 하둡 클러스터 셋업.
- 2008년 1000+ 노드 하둡 클러스터를 셋업
  - 웹페이지 그래프 계산을 하둡으로 포팅
- 2009년 30여개 마켓의 모든 검색어를 하둡에 저장하고 처리.
- 웹페이지 classification이나 Machine Learned Ranking등의 모델 빌딩에 하둡 클러스터 사용

야후 검색팀의 예 (하둡성숙기)

- 몇개의 하둡 성공 스토리 이후 하둡팀이 전사적인 조직으로 확대 (Platform 그룹).
  - 2011년 HortonWorks라는 회사로 스핀오프.
- 미디어 팀들을 포함한 거의 모든 팀들이 사용하기 시작
  - 하둡이 일종의 corporate-wide 데이터 저장소로 변신 -> Web Of Object 프로젝트.
  - 4개의 하둡 클러스터 존재.
    - 조직별, 리서치용 vs. 프로덕션용

몇가지 교훈들

- 대용량 데이터 중앙수집의 어려움
  - 빅데이터 처리를 위해서는 그 빅데이터를 한군데로 모으는 것이 시작인데 여러가지 어려움이 존재.
  - 소프트웨어 변경이 필요하며 여러 유관팀의 도움이 필요.
- 성공스토리의 필요성
  - 성공스토리가 있어야 보다 더 많은 팀의 adoption이나 매니지먼트의 지원을 끌어낼 수 있음.
- ROI를 고려
  - 데이터가 있다고 무작정 그걸 처리하려고 하기 보다는 무엇을 할 것인지 그게 리턴이 있을지 먼저 고려. 빅 데이터 처리 시스템을 만드는 것은 많은 시간과 비용이 들어간다는 점을 명심.
- 데이터 접근 민주화의 중요성
  - 그전에는 샘플조차 얻기 힘들던 데이터들이 접근도 되고 그걸 쉽게 처리할 수 있는 시스템까지 제공되자 크고 작은 이노베이션들이 쏟아져 나옴.

오픈소스가 대세?

빅데이터 시스템의 구성

데이터수집

- 빅데이터의 시작은 데이터를 수집하여 처리할 수 있는 장소로 올리는 것!
- 데이터의 용량이 큰 경우 데이터의 수집자체가 큰 문제.
  - 네트웍을 타고 업로드하는 자체가 오랜 시간이 걸림. 많은 경우 데이터발생소스와 하둡을 같은 데이터센터에서 고속네트웍으로 연결
- 몇가지 오픈소스 솔루션이 많이 쓰임
  - **Flume: Cloudera**에서 만들어서 지금 **Apache** 오픈소스.
  - **Chukwa: Apache** 오픈소스
  - 기본적으로 분산환경을 기반으로 여러대의 데이터소스로부터 데이터를 받아다가 계층구조 형태로 머징하는 형태의 구조를 갖고 있으며 데이터 푸시보다는 데이터 풀링을 많이 사용.

데이터저장과 처리

- 대부분의 빅 데이터 시스템에서 이 역할을 하는 것은 바로 하둡.
  - 데이터의 저장소이자 프로세싱 브레인.
  - 이 강의의 핵심.
- 프로세싱을 위해서 여러가지 언어가 만들어짐
  - Java MapReduce, Hive, Pig, Presto, ...
- 하둡을 기반으로한 생태계가 만들어지고 있으며 많은 회사들이 관련 소프트웨어/서비스를 만들고 있음
  - IBM, EMC/Greenplum, EMC/VMWare, Amazon, Microsoft, SAS, SAP, Cloudera, HortonWorks, MapR, ...

웍플로우 실행 및 관리

- 계속적으로 발생하는 데이터의 처리를 위해 처리작업들의 실행이 자동화되어야함
  - 복잡한 ETL 작업의 경우 수십개의 job들의 chaining이 필요.
  - 주기적으로 혹은 데이터가 특정 위치에 생기면 특정 Job을 시작하게 하는 메커니즘이 필요. 즉, 웍플로우 관리가 필요.
- 몇개의 오픈소스 프로젝트가 널리쓰임.
  - Oozie, Azkaban, Airflow, Pinball, ...

결과 데이터의 액세스

- 하둡으로 처리된 데이터는 어떤 형태로건 바깥에서 액세스가 필요
- 3가지 정도의 패턴 존재
  - RDBMS 혹은 Redis에 저장.
    - 작은 크기 데이터에 적합. 예를 들면 리포트.
  - NoSQL에 저장
    - HBase, Cassandra, MongoDB 등등.
    - 이 경우 데이터크기에 관계없이 액세스가능. Ad-hoc 분석을 위한 방법도 제공.
  - Search Engine에 저장
    - Lucene, Solr, ElasticSearch 등등.

데이터 Visualization

- 데이터를 어떻게 보기쉽고 이해하기 쉽고 멋있게 보여줄 수 있을까? InfoGraphics
- 어떤 데이터를 분석할때 처음 시작 작업은 다양한 형태로 그 데이터의 분포나 패턴을 그려보는 것이 중요
  - 데이터분석의 시작은 필요한 데이터가 수집되고 있는지 수집상에 오류는 없는지 검증하는 것 (data clean-up).

데이터 마이닝 – Data Scientist

- 일단 시스템이 구성되고 나면 누군가 데이터에서 새로운 가치와 의미를 찾아야함.
- Data Scientist의 몫
  - 수학/통계 지식 (모델링)
  - 프로그래밍 스킬
  - 데이터분석에 대한 열정과 비지니스에 대한 이해.
- Mahout, R 등이 널리 쓰임

성공 스토리들

Fraud Detection

- Bank of America, Chase등의 은행은 과거 신용카드의 과거 트랜잭션 데이터들을 바탕으로 fraud detection 모델을 빌딩.
- 모든 트랜잭션은 fraud detection 모델을 거침.
- 모델 빌딩은 빅데이터 시스템의 도움없이는 불가능.
  - 충분한 데이터의 수집.
  - 주기적인 모델의 빌드를 가능.
  - 빠른 실험과 테스트가 가능 **(개발기간의 단축)**

Netflix 영화 추천

- 25M+ subscriber, 30M movie play per day, 4M rating per day, 3M searches a day, 2B hours streamed in Q4 2011
  - 75% 영화감상이 영화 추천에 기반함.

- Markov chain기반의 알고리즘
  - 거대 NxN 행렬 계산.
  - 처음에는 RDBMS기반으로 일주일에 한번 주말에 실행.
  - Hadoop 도입이후 지금은 매일 한번씩 계산.
  - 성능상의 이유로 Netflix Prize 우승 알고리즘은 사용못함.

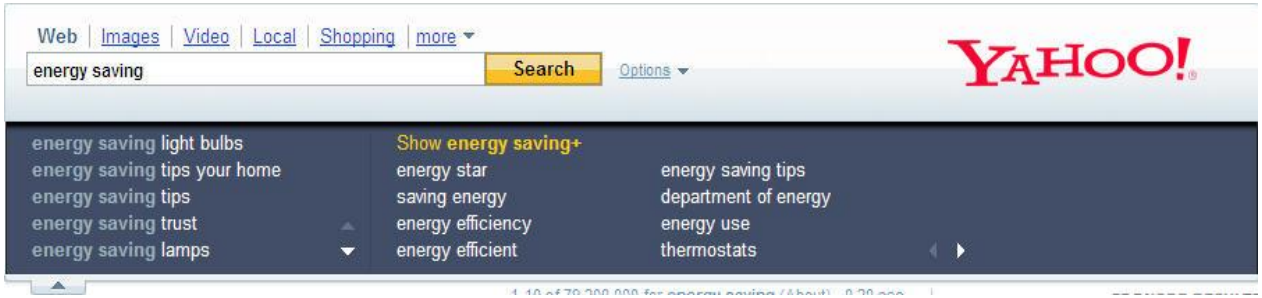
Bioinformatics - DNA 분석

- 인간의 유전체는 총 30억쌍. 1인당 DNA 정보는 대략 120GB.
- 하둡을 기반으로 DNA분석과 비교를 해주는 회사들이 등장하기 시작
  - Cloudburst, Crossbow, Hadoop-BAM, ...
  - 한국에서는 얼마전에 SDS에서도 서비스를 발표.

Trulia

- 부동산 가격 및 예측 사이트. 2006년 설립된 샌프란시스코 기반의 스타트업 (2013년 IPO 추진중)
- 부동산세 정보와 부동산 판매가격을 계속적으로 수집/조인하여 가격을 예측.
  - 처음에는 이 프로세스를 MySQL로 구현. 미국전체 데이터를 돌리는데 일주일 걸림.
  - Hadoop으로 포팅후 7시간으로 단축. 다양한 실험이 가능해짐.

Yahoo



- 검색어 자동완성 데이터베이스가 하둡을 이용해 빌딩됨.
  - 3 년치의 로그 데이터
  - 20여개의 MapReduce job들이 데이터베이스를 빌딩.

	Before Hadoop	After Hadoop
시간	26일	20분
개발언어	C++	Python
개발시간	2-3 주	2-3 일

문제점 (혹은 유념사항)

프라이버시 이슈

- 빅데이터 시스템의 등장은 그 전까지는 불가능했던 레벨의 **데이터 수집과 조인**을 가능케함 -> 디지털 빅브라더의 탄생이 가능.
- **EU**의 경우 선도적으로 많은 부분에서 규제장치를 도입
  - 검색엔진 쿼리의 경우 개인관련 정보(**IP**주소, 브라우저 쿠키 정보)를 **6**개월 이상 저장하지 않도록 권고.
- 예) 개인화된 검색어 자동완성의 문제점.

ROI

- MySQL 한대로 충분한지 먼저 생각. 하둡을 기반으로한 빅데이터시스템은 시간,돈,노력이 모두 많이 들어간다는 점을 명심.
- 정말로 스케일이 문제가 될 경우에만 고려.
- 고려시에도 처음부터 하드웨어부터 다 준비하지말고 클라우드 서비스를 이용해서 가능성 타진.

오픈소스로 구성된 시스템

- 요즘 빅데이터시스템들은 대부분 오픈소스 프로젝트들을 여러개 모아서 만들어지고 있음.
- 이는 문제와 함께 기회를 제공.
  - 보안 문제 가능성 (완전 공개된 소스).
  - 오픈소스는 굉장히 빠르게 진화하며 없어지기도함 (호환성 이슈, 버전간 충돌 이슈 등등).
  - 많은 스타트업들이 버전관리와 서포트를 해주는 배포판제공.

누가 빅데이터시장에서 알아야할 플레이어들인가?

관련 회사들

아파치 재단

- 비영리재단으로 빅데이터 시스템의 거의 모든 프로젝트들이 아파치재단의 오픈소스 프로젝트들. 기업스폰서십이나 개인들의 기부, 컨퍼런스 주최등으로 유지.
  - 현재 100개의 톱레벨 프로젝트가 존재.
- 아파치 라이센스는 상업적인 목적으로 사용하기에 제약이 거의 없는 라이센스.
- www.apache.org

Cloudera

- 2008년 설립된 하둡기반 빅데이터 스타트업. 가장 활발하고 유명하며 많은 수의 하둡관련 오픈소스 프로젝트에 참여.
- 여기서 만든 하둡 배포판이 가장 많이 사용됨 (CDH라 부름)
- Hadoop World라는 연례 컨퍼런스 주최.
- 하둡관련 교육과 컨설팅으로 주매출 달성.
- 하둡의 창시자 Doug Cutting도 2009년 조인

HortonWorks

- 2011년 야후내의 하둡플랫폼이 분사하여 설립됨. 2006년부터 야후내에서 하둡관련 일을 해온 EricBaldeschwieler가 CTO로 재직.
- 하둡의 초기 발전에 많은 공헌을 함.
- 하는 일이나 성격은 Cloudera와 굉장히 흡사. Hadoop Summit이란 연례 컨퍼런스 주최.

기타 플레이어들

- **MapR: Cloudera**나 **HortonWorks**와 비슷한 스타트업. 자체하둡배포판이 있음.
- **IBM, EMC/Greenplum, SAS, SAP:** 기존의 **RDBMS**나 **DW** 솔루션에서 하둡기반의 빅데이터로 선회한 케이스들.
- **EMC/VMWare:** 빅데이터 시스템들의 가상화라는 측면에서 접근.
- ....