*REPORT*

과 목 : 빅데이터와 데이터과학

담당교수 : 홍진근 교수님

학 부 : 컴퓨터공학부

학 과 : 소프트웨어학과

학 번 : 20191306

이 름 : 김대원

1. 데이터 처리 단계별 기술 :

데이터 소스 → 수집 → 저장 → 처리 → 분석 → 표현 순입니다.

2. 생성 :

2-1. 내부 데이터 : 자체적으로 보유한 내부파일 시스템, 데이터베이스 관리 시스템, 센서에서 정형 데이터를 수집합니다.

2-2. 외부 데이터 : 인터넷으로 연결된 외부 비정형 데이터를 수집합니다.

3. 수집 :

3-1. 로그 수집기 :

내부 웹서버 로그를 수집합니다.

3-2. 크롤링 :

웹 페이지 데이터를 추출합니다.

3-3. ELT :

소스 데이터를 취합해 데이터를 추출, 변환, 적재하는 과정을 지원합니다.

4. 센싱 :

각종 센서로 데이터를 추출합니다.

5. 저장 :

5-1. 분산 파일 시스템 : 네트워크로 공유하는 호스트 파일에 접근할 수 있는 파일 시스템입니다.

5-2. 병렬 DBMS : 다수의 마이크로프로세서를 사용하여 데이터베이스를 처리를 동시에 수행하는 데이터베이스 시스템입니다.

5-3. 네트워크 구성 저장 시스템 : 총괄적 데이터 저장 및 관리를 하는 시스템입니다.

6. 처리 및 관련 기술 :

6-1. 맵리듀스 : 분산 병렬 데이터 처리기술 표준, 대용량 데이터를 분산 처리하기 위해 만들어진 프로그램입니다.

6-2. 하둡 : 여러 개의 저렴한 컴퓨터를 마치 하나의 것처럼 묶어 대용량 데이터를 처리하는 기술입니다. 정형, 비정형 데이터를 분석합니다.

6-3. 카산드라 : 분산 시스템 대용량 데이터 처리 관리 시스템입니다.

6-4. Hbase : 분산 비관계형 데이터베이스입니다.

7. 분석 :

7-1. 텍스트 마이닝 : 자연어 처리기술을 이용하여 비정형 텍스트에서 유용한 정보를 추출, 다른 데이터와 연계성 파악, 분류나 군집화 등 빅데이터에 숨겨진 정보를 발견합니다.

7-2. 웹 마이닝 : 인터넷 정보를 데이터 마이닝 기법으로 분석합니다.

7-3. 오피니언 마이닝 : 콘텐츠에서 표현된 의견을 추출, 분류, 이해, 자산화 하는 것 입니다.

7-4. 리얼리티 마이닝 : 인간관계와 행동 양태 등을 추론 합니다.

7-5. 군집화 : 특성이 비슷한 테이터를 합쳐 하나의 군으로 묶어 분류합니다.

8. 분류 :

훈련가능한 데이터군을 학습시켜 새로 추가되는 데이터가 속할 만한 데이터군을 찾는 학습방법입니다.

9. 표현 :

9-1 : 획득 : 테이터 획득, 재해석을 합니다.

9-2 : 가시화 : 데이터를 도표나 그래픽으로 표현합니다.

10. RDBMS :

장&단점 : 데이터 무결성, 정확성 보장, 정규화 된 테이블과 소규모 트랜잭션이 있음, 확장성에 한계가 있음, 클라우드 분산 환경에 부적합

특성 : update, delete, join 연산 가능, ACID 트랜잭션이 있음, 고정 스키마가 있음

11. NoSQL :

장&단점 : 데이터의 무결성과 정확성을 보장하지 않음, 웹 환경의 다양한 정보를 검색, 저장 가능

특성 : 수정, 삭제를 사용하지 않음(입력으로 대체), 강한 일관성은 불필요, 노드의 추가 및 삭제, 데이터 분산에 유연

기술적 특성 :

1. 無스키마 : 데이터를 모델링하는 고정된 데이터 스키마 없이 키 값을 이용하여 다양한 형태의 데이터 저장 및 접근 가능, 데이터 저장 방식은 크게 열, 값, 문서, 그래프 등의 네 가지를 기반으로 구분

2. 탄력성 : 시스템 일부에 장애가 발생해도 클라이언트가 시스템에 접근 가능, 응용 시스템의 다운 타임이 없도록 하는 동시에 대용량 데이터의 생성 및 갱신, 질의에 대응할 수 있도록 시스템 규모와 성능 확장이 용이하며, 입출력의 부하를 분산시키는 데도 용이한 구조

3. 질의 기능 : 수십 대에서 수천 대 규모로 구성된 시스템에서도 데이터의 특성에 맞게 효율적으로 데이터를 검색, 처리할 수 있는 질의 언어, 관련 처리 기술, API 제공

4. 캐싱 : 대규모 질의에도 고성능 응답 속도를 제공할 수 있는 메모리 기반 캐싱 기술을 적용하는 것이 중요, 개발 및 운영에도 투명하고 일관되게 적용할 수 있는 구조

12. S3 :

최소 1바이트에서 5테라바이트 데이터가 포함된 객체의 읽기, 쓰기, 삭제가 가능하며, 저장 기능한 객체 수에 제한이 없음. 각 객체는 버킷에 저장, 개발자가 할당한 고유한 키로 검색 가능

13. HDFS :

병렬 처리를 지원하고 사용자에게 모든 처리 과정을 추상화한 인터페이스 제공. 하둡은 마스터 하나와 슬레이브 여러 개로 클러스터를 구성

14. DynamoDB :

NoSQL 데이터베이스. 하드웨어 프로비저닝, 복제, 설정 패치, 사용하는 응용 프로그램에 따른 DB 자동 분할 기능 등 지원

15. MongoDB :

신뢰성과 확장성에 기반한 문서 지향 데이터베이스. 저장의 최소 단위는 문서. RDBMS의 테이블과 비슷한 컬렉션이라는 곳에 수집.

16. CouchDB :

인터페이스가 자바스크립트로 구성. MongoDB와 비슷하지만, 제공 질의, 확장성, 버전 관리 등에서 성능이 더 우수. 다중 버전 동시 동작 제어 기능으로 해결. 쓰기 작업 중 충돌이 발생하면 다른 기술에 비해 클라이언트의 부하가 증가한다는 단점.

17. Cassandra :

<키, 값> 구조의 DBMS. 계층적 열구조를 논리 구조의 배경으로 함. 인박스 환경 때문에 개발한 만큼 방대한 양의 데이터 저장과 처리, 정렬에 활용되는 속성 제한, 빈번한 데이터 변경에 사용하기 좋은 환경.

18. HBase :

행, 열 그룹, 열 이름, 타임스탬프를 이용한 테이블 구조로 되어 있음. ZooKeeper를 노드 관리에 사용. Get/Put/Scan/Delete의 네 가지 동작을 지원.

19. Redis :

메모리 기바느이 <키, 값> 저장 공간. 다양한 데이터 구조를 지원. 메모리에 저장된 내용을 지속시키려고 파일로 싱크하는 기능 제공.

20. Riak :

데이터 모델 <키, 값> 저장 형식을 취함. JSON문서가 저장되는 문서 지향 데이터베이스 형식. 링 형태의 데이터 분산 저장 구조로, 해시 알고리즘과 데이터 키에 따라서 적정 노드를 찾아가는 구조.

21. Hypertable :

열 그룹과 티임스탬프 개념을 사용. C++로 개발. 여러 언어 지원. 다양한 분산 파일 시스템을 기반으로 구성.

22. ZooKeeper :

하둡의 분산 처리 시스템을 관리하는 분산 처리 시스템을 일괄적으로 관리하는 시스템. 네임 서비스를 이용한 부하 분산, 분산 락이나 동기화 문제 해결, 장애 상황 판단 및 복구 등의 특징.

23. Voldemort :

Java 기반 <키, 값> 저장 형식인 데이터베이스 일종의 해시 테이블로 볼 수 있음. 복잡한 구조의 백엔드로, 분산 대형 클리스터 환경에서 동작. 일관성 있는 해싱을 사용하여 키와 관련된 저장 위치를 빠르게 조회, 버전을 제어하여 일치하지 않는 값들을 신속하게 처리.

\*추상화 : 복잡한 자료, 모듈, 시스템 등으로부터 핵심적인 개념 또는 기능을 간추려 내는 것.

24. Hadoop :

엔진 형태 미들웨어와 소프트웨어 개발 프레임워크로 구성. 맵리듀스의 분산 처리 구조. 맵리듀스는 하나의 큰 데이터를 여러 개의 조각으로 나누어 처리하는 맵 단계와 처리된 결과를 하나로 모아서 취합한 후 결과를 도출해 내는 리듀스 단계로 구성.

25. Pig :

절차적 데이터 처리 언어 프레임워크. 대규모 데이터 처리가 용이. 데이터 흐름을 명시적으로 보여 줄 수 있는 코드 작성이 가능, 유지보수 용이. 코드 실행을 자동으로 최적화함. Int, long, double등 기본형 외 릴레이션, 백, 튜플과 같은 고수준의 구조를 제공, Filter, Foreach, Group, Join, Load, Store등 관계 연산도 지원.

26. Hive :

SQL과 같이 선언적으로 데이터를 처리. 맵리듀스 기반의 실행 부분, 데이터가 저장된 공간의 메타데이터 정보, 사용자나 응용 프로그램에서 질의를 입력받아 실행시키는 실행 부분으로 구성. 스칼라 값, 집합, 테이블 수준에서 사용자 정의 함수를 지정할 수 있는 기능 제공.

27. Cascading :

하둡용 오픈 소스 소프트웨어 추상화 계층. JVM기반의 언어를 활용. 맵리듀스 작업의 복잡성을 숨겨줌. 주요 특징에는 데이터 처리 API 제공, 데이터 통합 API 제공, 프로세스 스케줄러 API 제공, 기업 개발 지원, 다양한 언어 지원, 내결함성 지원 등이 있다.

28. Cascalog :

Clojure 언어로 개발 기능적인 데이터 처리 인터페이스를 제공. SQL보다 훨씬 높은 수준의 사용자 추상화를 제공. 기능, 필터, 집계가 모두 동일한 구문을 사용해 작성이 간편. Null 값의 오류를 최소화.

29. Mrjob :

데이터를 처리하는 코드를 작성 한곳에서 작동하는 엘라스틱 맵리듀스나 하둡 클리스터에서 동작하는 프레임워크를 제공.

30. S4 :

오픈 소스 기반의 분산 스트림 처리 시스템. ZooKeeper 프레임워크 사용 클러스터들의 분산 실행을 관리. 완전히 균형적인 분산 시스템.

31. MapR :

HDFS를 대체할 수 있는 자체 파일 시스템으로 운영. 네트워크 파일 시스템과 호환할 수 있는 데이터 전송의 단순화를 제공하여 맵리듀스 처리량을 향상.

32. Acunu :

Castle은 커널 레벨에서 <키, 값>을 저장함으로써 속도를 향상, 기존의 관계형 데이터베이스와 비슷한 자동 구성과 다양한 관리 기능을 제공. 제어 및 모니터링 등이 포함된 관리 도구를 제공.

33. Azkaban :

각 서비스가 여러 개의 연산들을 통합하여 비즈니스 로직으로 실행. 규모가 큰 프로세스들은 50단계 이상의 세부 프로세스로 구성 가능.

34. Oozie :

워크플로우를 나타낼 수 있는 다양한 언어를 지원. Java 서블릿 컨테이너에서 실행하는 Java 웹 응용 프로그램으로, 사용자가 시스템의 기능을 확장할 수 있는 API도 지원. 제어 플로우 노드는 워크플로우의 시작과 끝(start, end, fail 노드)을 정의한다.

35. Greenplum :

Shared-Nothing MPP 구조의 데이터베이스. 저장 데이터는 적용되는 연산에 따라 행 기반이나 열 기반 방식 중 하나를 선택할 수 있음. 병렬 질의나 맵리듀스 형태의 프로그램을 효율적으로 처리. Java와 R 언어로 쓰인 사용자 함수를 지원.

36. EC2 :

클라우드에서 컴퓨팅의 규모를 자유롭게 변경할 수 있는 웹 서비스. 컴퓨팅 리소스 전체를 제어할 수 있는 권한을 제공하며, 아마존의 검증된 컴퓨팅 인프라에서도 실행가능. 메모리 및 CPU 자원을 시간 단위로 지정하여 컴퓨터를 임대할 수 있음.

37. Heroku :

다양한 프로그래밍 언어를 지원하는 클라우드 개발 도구 중 하나. 사용자가 서버 관리를 신경 쓰지 않고 응용 프로그램 개발과 관리에만 집중 가능.

38. R 프로그래밍 언어 :

통계적으로 계산하고 그래픽을 처리하는 프로그래밍 언어이자 소프트웨어 환경. 다양한 통계 기법과 수치 해석 기법을 지원, 사용자가 제작한 패키지를 추가하여 기능을 확장. 데이터 분석 결과를 시각화 할 수 있음.

39. Pipes :

사용자가 컴포넌트를 드래그 앤 드롭하는 방식의 그래픽 인터페이스를 사용.

40. Mechanical Turk ; Mturk :

컴퓨터가 수행할 수 없는 임무를 수행하게 하려고 프로그래머가 사람의 인지 능력을 사용하는 것을 도와주는 크라우드 소싱 인터넷 마켓 플레이스이다.

41. Solr/Lucene :

Lucene 기반의 기업 검색 플랫폼인 Solr는 Java로 작성. 아파치 톰캣과 같은 서블릿 내에 있는 독립적인 전체-텍스트 검색 서버로 작동. Solr는 파일을 인덱스하는 검색 엔진 이상으로, XML 요청을 HTTP로 보내는 웹 서비스 API가 있는 검색 서버.

42. ElasticSearch :

오픈 소스 검색 엔진 서비스. 일반 사용자에게 적합한 서비스를 제공. 문서 지향적, 신뢰, 비동기 쓰기, 실시간 검색 가능, 설치 및 사용 쉬움. 하나의 클리스터에 문제가 발생하면 진행 중인 작업이 복제 되어있는 다른 클리스터로 자동으로 이동하여 결과를 보여 줌으로써 단일 고장점을 제거.

43. Datameer :

단순한 프로그래밍 환경 제공. 하둡의 복잡성을 숨긴 채 분석 도구를 제공.

44. InfoSphere BigInsights :

인터넷에서 정형화된 데이터 및 비정형화된 데이터를 관리하고 분석하는 IBM의 다목적 솔루션. 관리, 워크플로우, 프러비저닝, 보안 기능을 접목.

45. InfoSphere Streams :

짧은 시간 안에 방대한 양의 스트리밍 데이터를 시속적으로 분석할 수 있는 IBM의 또 다른 솔루션. 다양한 정형화 데이터 및 비정형화 데이터 유형을 모두 지원하는 강력한 확장성과 빠른 속도의 인프라를 제공.

46. 데이터 마이닝 :

데이터 마이닝은 저장된 방대한 양의 데이터 안에서 자동으로 체계적이고 통계적인 규칙이나 패턴을 찾아내는 데이터베이스 파생 기술이다. 데이터 마이닝 기법은 통계학 분야에서 발전한 탐색적 데이터 분석, 가설 검증, 다변량 분석, 시계열 분석, 일반선형모형 등 방법론과 데이터베이스 분야에서 발전한 OLAP, 인공지능 분야에서 발전한 SOM, 신경망, 전문가 시스템 등 기술적 방법론 등에 사용한다.

46-1) 분류 : 일정한 집단에서 특정한 정의를 이용하여 분류 및 구분을 추론한다. 예를 들어, 경쟁자에게로 이탈한 고객을 분류해 내는 기술을 들 수 있다.

46-2) 예측 : 방대한 양의 데이터 집합의 패턴을 기반으로 미래를 예측한다. 예를 들어, 수요를 예측하는 기술을 들 수 있다.

46-3) 시계열 분석 : 시간의 변화에 따라 일정한 간격으로 연속적인 통계 숫자를 저장한 시계열 데이터에 바탕을 둔 분석 방법이다. 예를 들어, 매일 주식의 값을 저장하는 시계열 데이터를 분석하는 기술을 들 수 있다.

46-4) 회귀분석 : 하나 이상의 변수 간의 영향이나 관계를 분석 및 추정하는 기술을 들 수 있다.

46-5) 군집화 : 구체적인 특성을 공유하는 군집을 찾는다. 군집화는 미리 정의된 특성의 정보가 없다는 점에서 분류와 다르다. 예를 들어, 비슷한 행동 집단을 구분해 내는 기술을 들 수 있다.

46-6) 연관 규칙 : 동시에 발생한 사건 간의 관계를 정의한다. 예를 들어, 장바구니 안에 동시에 들어가는 상품들의 관계를 규명하는 기술을 들 수 있다.

46-7) 요약 : 데이터의 일반적인 특성이나 특징의 요점을 간략히 정리하는 기술을 들 수 있다.

46-8) 연속성 : 시간에 따라 순차적으로 나타나는 사건의 종속성을 말한다. 예를 들어, A 제품을 구입한 고객이 향후 B 제품을 구입할 확률이라든가 작년의 계절적 매출 변동 요인과 올해의 매출 등을 알아내는 기술을 들 수 있다.

47. 텍스트 마이닝 기술 :

텍스트 마이닝은 비, 반정형 텍스트 데이터로 구성된 빅데이터에서 자연어 처리 기술에 기반하여 의미 있는 정보를 추출하는 기술. 텍스트 마이닝의 분석 대상은 텍스트 문서, 이메일, HTML 파일 등과 같은 비, 반정형의 텍스트 데이터라는 차이점이 있다.

텍스트 마이닝 과정은 구조화되지 않은 형태의 문서를 수집하면 전처리과정을 거쳐서 텍스트분석을 수월한 형태로 변환.

전처리작업은 텍스트를 분석하는 알고리즘의 성능을 결정짓는 중요한 요소이다. 전처리를 거친 텍스트 데이터는 컴퓨터가 처리할 수 있는 다양한 방식으로 표현, 일반적으로 문서의 단어들을 벡터 공간속에서 표현. 전처리과정을 거친 데이터들을 대상으로 정보 추출, 범주화, 문서 요약 등 다양한 방식으로 텍스트를 분석한다.

47-1) 정보 수집 : 비, 반정형의 텍스트 데이터를 수집하는 단계이다.

47-2) 정보 처리 : 대용량의 데이터에서 특정 키워드나 일부 의미 있는 요소를 추출하려고 전처리를 하는 단계이다.

47-3) 정보 추출 : 수학적인 모델이나 알고리즘을 이용하여 유용한 정보를 추출해 낸다. 텍스트 마이닝을 위한 정보 추출 방법에는 다양한 목적, 조건, 환경 등이 있는데, 이 정보 추출 방법은 텍스트 마이닝에서 가장 중요한 부분 중 하나이다. 특히 정보 추출 방법에는 수많은 수학적 알고리즘과 방법이 있으며, 그중 간단하면서 가장 강력한 방법인 TF-IDF방식을 많이 사용한다.

47-4) 정보 분석 : 최종 키워드나 의미 있는 요소의 우선순위를 도출하는 단계이다.

\* TF-IDF란? 정보 검색과 텍스트 마이닝에서 이용하는 키워드의 가중치를 구하는 방법이다. 여러 문서로 된 문서군이 있을 때 어떤 단어가 특정 문서 내에서 얼마나 중요한지 나타내는 통계적 수치이다.

데이터 분석 방법론을 크게 4단계로 나누면?

1. 전처리 : 분석을 위한 적당한 포맷으로 데이터를 전처리

2. 분석 : 중요한 개념과 용어를 텍스트 마이닝으로 추출

3. 개선 & 이해 : 추출된 개념과 용어의 패턴 분석

4. 보고 & 기록 : 분석된 결과 보고

48. 오피니언 마이닝 기술 :

오피니언 마이닝 기술은 빅데이터에 포함된 어떤 사안이나 인물, 이슈, 이벤트에서 사람들의 의견이나 평가 등을 분석하는 것. 불특정 다수 사용자의 오피니언으로 된 데이터에서 특정 주제, 대상과 관련된 의견이나 선호도를 빠르게 파악할 수 있어 많은 분야에서 사용.

일반적으로 오피니언 마이닝 과정은 특징 추출, 의견 분류, 요약 및 전달 3단계로 진행

48-1) 특징 추출 : 의미 있는 요소, 특징을 추출하는 단계이다.

48-2) 의견 분류 : 어떤 의미로 사용되었는지 분석하고 분류하는 단계

48-3) 요약 및 전달 : 선호도가 밝혀진 오피니언 정보를 요약하고 사용자에게 전달하는 단계

오피니언 마이닝에서 긍정과 부정의 의견을 분류하는 과정을 구체적을 표현하면?

1. 긍정 및 부정을 표현하는 단어 정보 추출 :

기존에 구축된 사전 등 리소스를 이용하거나 수작업으로 해당 도메인의 고빈도 긍정/부정을 표현하는 단어들을 확인할 수 있음. 학습 데이터에서 유용한 통계 정보를 활용하여 자동으로 어휘 정보를 얻을 수 있음. 통계적인 방법을 적용하여 어휘 정보를 추출할 때 상호 정보량과 같은 평가 척도를 사용할 수 있음. 한국어로 작성한 사용자 별점이 아주 높(4~5점)은 리뷰에서는 고빈도 단어를 긍정 표현으로, 사용자 별점이 아주 낮은(1~2점) 리뷰에서는 고빈도 단어를 부정으로 표현.

2. 세부 평가 요소와 그것이 가리키는 오피니언의 연결 관계를 포함한 문장 인식 :

첫 번째 단계에서 구축된 어휘 정보를 사용하여 세부 평가 요소와 긍정/부정 표현을 찾음. 규칙 기반 방법과 통계 기반 방법을 동시에 사용할 수 있음.

3. 긍정/부정 표현의 수 및 유용한 문장을 추출하여 리뷰 요약 생성 :

각 세부 평가 요소에서 긍정 표현과 부정 표현의 차이를 이용하여 사용자들의 선호도를 제시할 수 있다. 또한 세부 평가 요소와 관련된 오피니언을 포함하는 문장 중 유의미한 문장들을 긍정/부정 평가별로 추출하여 중요 문장으로 구성된 리뷰 요약을 생성할 수 있음. 오피니언 마이닝의 결과는 긍정/부정 평가의 정도를 나타내거나 요약 형태로 제시할 수 있음.

49. 리얼리티 마이닝 기술 :

리얼리티 마이닝은 데이터 마이닝 기술의 일종으로, 모바일 기기, GPS 등의 사용 형태에 기반하여 사용자의 인간관계, 행동 방식, 라이프 스타일 등을 분석한다. 다른 마이닝 기술과는 다르게 생체 인식으로 소비자에게 맞춤화 된 마케팅이나 사용자의 감정에 따른 심리적 변화를 추적하여 제품 판매 등에 반영함.

생체 인식은 개개인이 지닌 고유한 생체적 특성을 이용하여 사람을 식별해 내는 기술.

생체 데이터 마이닝의 수행 과정은 데이터, 정보, 지식, 결정의 4단계로 구성.

1. 데이터는 선택과 데이터 정제를 담당.

2. 정보는 데이터 변환을 담당. 즉, 데이터 변환은 데이터베이스에 이미 있는 튜플에서 의미 있는 새로운 데이터로 사용할 수 있게 더 많은 정보를 포함하도록 몇 개의 튜플을 하나의 튜플로 변환하는 작업을 수행.

3. 실제로 적용 가능한지를 평가하는 단계로, 피드백 과정을 거쳐 데이터 마이닝 과정을 반복해서 재수행함.

4. 결정 단계는 서버에서 응급 상황과 건강 정보를 구분하여 응급 상황이면 서비스 대상자의 최단 거리에 위치한 이웃 주민과 해당 기관의 복지 담당자에게 음성 호출 서비스를 시행. 그 후 데이터베이스에 축적된 건강 정보를 이용하여 웹 기반의 건강 관리 서비스를 수행.

50. 군집화 기술 :

군집화 기술은 데이터 마이닝 기술의 한 방법으로, 주어진 빅데이터에서 데이터들의 특성을 고려하여 군집을 정의하고 군집을 대표할 수 있는 대표점을 찾는 것. 군집은 특성이 비슷한 데이터들로 구성한 집단을 말하므로, 데이터의 특성이 다르면 군집에 속해야 한다. 군집화 기술은 크게 계층적 군집화 기술과 분할적 군집화 기술로 나눌 수 있음.

계층적 군집화 방법은 먼저 각 데이터 점을 하나의 군집으로 설정한 후 이들 간의 거리를 기반으로 하여 분할, 합병해 가는 방식이다. 이렇게 상향식 방식으로 모든 점이 하나의 대형 군집에 속할 때까지 수행함. 군집에 속하는 히스토리 정보는 유지 됨. 이것은 응집하는 계층적 군집화 기술이라고도 함. 가까운 데이터끼리 군집화하는 방법임.

이 알고리즘은 우선 데이터를 n개로 가정하고, 이 n개의 데이터를 서로 다른 n개의 다른 그룹으로 가정한다. 그룹 간의 유사성을 계산한 후 가장 유사도가 높은 두 그룹을 병합하여 그룹 개수를 줄이는 방법이며 이 과정을 전체 그룹의 수가 k개가 될 때까지 반복하여 k개의 그룹을 찾음.

계통도에서는 각 계층에서 군집들의 유사성을 쉽게 확인할 수 있음. 계층적 군집화는 크게 흡수와 분리 처리 과정으로 나뉨. 구조는 이진 트리 구조로 처리됨.

분할적 군집화 기술은 k개의 분할 영역을 결정하는 방법으로 유클리디안 거리 계산법에 기반함. 군집의 계층은 고려하지 않고, 군집을 평면적으로 분석하는 방법임. 이 군집은 군집 상태를 나타내는 척도 함수가 더 이상 변하지 않을 때까지 반복되며, 더 이상 변하지 않으면 그 상태의 그룹들을 군집화의 결과로 정함.

K-means 알고리즘 수행하는 과정은?

1. 군집의 개수인 k를 결정하고 각 군집에 초깃값으로 중심 한 개씩을 할당하여 위치를설정.

2. 각 데이터를 주어진 중심점을 기준으로 가장 가까운 군집에 할당. 중심점과의 거리는 유클라디안 거리 계산 방법에 따름.

3. 할당된 데이터를 중심으로 각 군집은 새로운 중심점을 계산함.

4. 새로운 중심점이 기존의 중심점과 차이가 있으면 2번으로 되돌아가 반복함. 새로운 중심점이 기존의 중심점과 차이가 없으면 알고리즘은 끝남.

51. 소셜 네트워크 분석 기술 :

네트워크는 사람을 연결하여 하나의 집단이나 조직을 만들고 , 이들을 다시 서로 연결시키는 일종의 ‘관계의 묶음’으로 생각할 수 있음. 사회적 구조체로 정의 가능. 특정한 네트워크를 분석하는데 일반적으로 사용하는 방법은 중심성과 연결성을 평가하는 것 임. 일반적인 접근 방법은 네트워크 그래프 구조를 대규모 행렬로 구성하고, 고유 벡터, 고유 값 계산을 포함한 다양한 행렬 연산을 수행하는 방식으로 진행.

최근에는 대규모 소셜 미디어 및 동적 소셜 네트워크에서 분석을 수행하는 방법으로 시멘틱 기술을 사용 함. 이 방법의 특징은 복잡하고 이질적인 그래프 구조를 트리플로 구성된 RDF 그래프로 표현하며 RDF(S) 기반의 네트워크 표현은 9가지 장점이 있음.

1. 표준 기반으로 의미의 모호성이 없는 정보 표현과 상호 호환성 확보가 가능함.

2. 상호 이질적 정보들을 단일한 표현 체계로 통합 표현, 연계가 가능함.

3. 그래프 구조뿐 아니라 각 노드의 특성 정보를 통합 표현 가능함.

4. 단일 서버에서 대규모 그래프로 표현, 저장, 관리가 가능함.

5. 그래프 구조에서 강력한 질의와 복잡한 데이터 연산 처리가 가능함.

6. URL에 기반을 두어 분산 그래프 DB 구현과 분산 질의 처리에 유리함.

7. 동적으로 변경되는 그래프 정보의 실시간 적용과 질의, 연산이 가능함.

8. 온톨로지 및 규칙과 결합하여 연역적 논리 추론이 가능함.

9. 기계 학습과 연동하여 하이브리드 유형의 분석 체계를 구현 할 수 있음. 특히 RDF(S)를 사용하므로 사람의 연결 구조와 정보 연결 구조를 통합하여 표현하고, 필요에 따라 부분적으로 분리하여 단순화하기도 쉬운데, 이는 소셜 미디어에 기반을 둔 소셜 네트워크 분석에서 매우 강력한 장점이 될 수 있음.

52. 그래프 마이닝 기술 :

그래프 마이닝 기술은 페이스북이나 트위터 같은 소셜 미디어의 데이터를 표현하는 방법이자 다양한 곳에 활용되는 매우 중요한 기술로, 빅데이터 중 가장 큰 빅데이터에 속한다.

그래프 마이닝은 데이터 마이닝 기술의 일종으로, 그래프에서 마이닝 기술을 적용하는 기술이다.

먼저 그래프는 노드(Node)와 에지(Edge)로 구성된 구조이다. 노드는 데이터 요소에 해당하는 정점이며, 에지는 노드간의 관계를 표시하는 간선이다. 그래프는 데이터 간의 관계를 표현할 때 사용하는 데이터 구조이다. 사용자가 관심 있는 최소 지지도 임계 값을 제시하면, 그것과 같거나 그 이상으로 빈도수가 높은 서브 그래프 패턴을 찾아낸다.

53. 소셜 네트워크 통계방안

1. 원 데이터 에지로만 표현

Ex) <1,2>, <1,3>, <2,3>, <3,1>

2. 각 맵 함수에서는 나누어진 에지의 <키, 값> 순서쌍에서 얻은 값인 목적 노드

Ex) ID로 그룹핑 : 컴바이너가 내장된 리듀스 함수로 결합된 결과를 보면, 1은 한 번, 2는 한 번, 3은 두 번 이라는 결과

3. 출력 정보를 다시 <키,값> 순서쌍으로 표현

Ex) <1,1>, <1,2>, <3,2>

맵 함수에서는 노드 전입 차수 과정을 그대로 반복

4. <키, 값> 순서쌍에서 얻은 값인 ‘해당 노드 전입 차수’로 그룹칭

Ex) ‘해당 노드 전입 차수’ 1은 두 번, 2는 한 번

54. 네트워크 분석 기법

1. 군집화 상수 :

네트워크에서 노드들(컴퓨터들)이 군집 정도가 얼마나 강한지 측정하는 방법

> 사람이나 컴퓨터의 관계 응집도를 평가하는 척도로 활용

2. 군집화 상수 함수 표현 :

2-1. cc(v) 식에서 분모는 노드 v의 모든 차수 즉, 가능한 모든 이웃의 크기(dv)를 2로 컴비네이션 하므로 노드 v의 가능한 모든 노드 쌍의 수를 분모

2-2. cc(v) 식에서 분자인 감마(r)는 노드 v의 아웃 즉, v에 연결된 노드들의 집합으로 분자의 결과인 <u, w>의 집합은 노드 v에 연결된 모든 이웃의 쌍인 실제로 연결된 에지 개수

3. 군집화 상수 표현 :

3-1. 군집화 상수가 높을수록 그래프 내의 노드 간에 관계성이 많음

> 군집화 상수는 사회학에서도 많이 활용

> 군집화 상수가 높을수록 사회적 신뢰도가 높음

3-2. 알고리즘 : 각 노드 v에 인접한 삼각형의 개수를 세어 군집화 상수를 구함

> 순차적인 방

> 그래프가 매우 커지면 연산의 속도가 떨어져 결과를 보기 어려움

> 빅 그래프에서는 맵리듀스로 바꾸어 처리해야 함

55. 맵 리듀스 :

맵 : 입력 데이터를 여러 개의 블록으로 분할하고, 각 블록에 대해 Apriori 알고리즘 수행 -> 빈발 아이템셋 추출

리듀스 : 맵 단계에서 추출한 빈발 아이템셋을 하나로 합쳐서 전체 빈발 아이템셋을 생성 (전체 연관규칙 생성)

56. 이동경로 예측 알고리즘 :

1. 데이터 분할 : 대용량 데이터를 여러 개의 블록으로 분할 -> 블록은 여러 개의 아이템으로 구성

2. 그리드 생성 : 블록들을 Apriori 알고리즘을 수행하는 노드들에게 할당

3. 빈발 아이템셋 추출 : 각 노드에서 자신에게 할당된 블록에서 Apriori 알고리즘을 수행하여 빈발 아이템셋 추출

4. 빈발 아이템셋 결합 : 각 노드에서 추출한 빈발 아이템셋을 결합하여 전체 빈발 아이템셋을 생성

5. 연관규칙 추출 : 전체 빈발 아이템셋을 이용하여 연관규칙을 추출, Apriori 알고리즘의 신뢰도 값 이용하여 연관규칙 필터링

57. 빅데이터와 분류기술 :

NN : 계산 노드 수

BN : 기본 분류기 수

PT : MReC4.5의 분할 수

MT : 맵 함수 수

RT : 리듀스 함수 수

TT : MReC4.5 실행의 총 소요 시간

58. 빅데이터와 군집화 :

1. 맵 과정에서 입력된 점에서 가장 가까운 점을 찾은 후

2. 리듀스 과정에서 이들을 이용하여 새로운 군집을 계산하고

3. 이후 모든 군집의 내용이 더 이상 바뀌지 않을 때까지 반복

59. 군집화에서 리듀스 함수 :

1. 서브 스페이스 군집화 알고리즘

- 서브 스페이스 군집화를 맵리듀스 프레임워크로 구현하여

- 다수 대의 머신에 데이터를 분산시킨 후

- 각 머신이 주어진 데이터만을 계산하도록 군집화

- 기본적인 군집화 알고리즘은 Apriori 방식 알고리즘을 이용하여

- 낮은 차원에서 밀집 영역을 찾아 나가는 방식을 이용

60. 빅데이터 주요 분석 기술 :

60-1. NLTK :

- 특정한 문제를 해결하려고 알고리즘 생성 과정에서 필요한 빌딩 블록 제공

- PHP, Python 등으로 제공되는 툴킷 들

- 많이 사용하는 프레임워크(사용자들이 다양한 방식으로 활용)

- 일반적인 접미사를 제거하여 중심 단어들을 추출하거나

- 전체 텍스트에서 동의어를 찾고자

- 기계가 분석 가능한 사전 형태로 데이터를 정형화시키는 기능 제공

60-2. OpenNLP :

- 비정형화된 텍스트에서 사람이나 기관의 이름, 특정 장소, 시간 등을 추출하는 작업을 간편하게 실행할 수 있는 모델을 포함하는 라이브러리

- Java 기반의 자연어 처리 솔루션

- 프로그래밍 언어 기반의 응용 프로그램에 적합한 형태

- 자연어 처리 코드를 스스로 생성

- 비정형화되거나 가공되지 않은 텍스트를 문장과 단어로 분리한 후

- 그 결과를 다양한 방법으로 클래스화하는 표준 컴포넌트 많이 포함

60-3. Boilerpipe :

- 웹 페이지 내에서 불필요한 부분을 제거하여 실질적으로 필요한 정보만 추출하는 작업을 수행하는 프레임워크

- HTML 문서에서 실제로 가장 중요한 콘텐츠를 찾아내는 알고리즘을 적용한 Java 기반 라이브러리로 제공

- 모든 종류의 웹 콘텐츠에서 좋은 전처리 도구 생성

- 뉴스나 기고 같은 특정 정보가 담긴 웹 페이지의 분석에 많이 사용

- 최근 블로그나 SNS 서비스 등 다른 형태의 웹 페이지에서 정보를 추출 유용

60-4. WEKA :

-데이터 분석 프로그램, 무료 배포, Java로 작성된 오픈 소스 프로그램

- 데이터마이닝 알고리즘으로 사용자가 개인화가 가능

- 플러그인 형식

- 개발자가 자신이 개발한 고유 알고리즘을 쉽게 접목가능

- 사용이 간편한 명령으로 구성

- 이해하기 쉽고 간단한 인터페이스를 제공

60-5. Mahout :

- 확장 가능한 기계 학습 알고리즘 제작

- 아파치 라이선스가 있으면 무료로 사용가능

- 방대한 양의 데이터 집합에서 기계 학습 알고리즘을 실행할 수 있는 오픈 소스 프레임워크, 확장성과 처리량을 보장하려고 하둡 기반의 병렬 형식으로 구성

- 공통 작업이 많은 알고리즘에 적합, 군집화, 분류, 사용자 행동을 기반으로 품목을 추천해 주는 시스템 등 분석환경에 적합

60-6. scikits\_learn :

- 서포트 백터 머신, 로지스틱 회귀분석, 군집화 등 여러 분석 기법 제공

- 고수준의 인터페이스 제공

- 복잡한 문제를 해결하는 알고리즘이나 문제를 해결하는 과정에 초점

61. 빅데이터 주요 표현 기술 :

61-1. Tag Cloud :

- 메타데이터에서 얻은 태그들을 중요도, 빈발도, 인기도 등 주제에 따라 분석하여 시각적으로 늘어놓은 것

- 대부분 웹 페이지나 이미지로 나타내며, 태그가 2차원 표와 같은 형태로 배치

- 특징적인 하나의 연결에 연관된 태그가 얼마만큼 많은지, 연관 태그들이 어떤 종류인지 시각화

61-2. Gephi :

- 제나 가공하지 않은 그래프 데이터를 네트워크 형태로 생성하여 이를 시각화하는 오픈 소스 기반의 Java 응용 프로그램

- 소셜 네트워크와 관련된 정보를 쉽게 이해 할 수 있도록 지원

- Likedln의 데이터를 표현하는 데 많이 사용

- 표현에 필요한 변수를 구성하려고 여러 알고리즘을 사용

- 사용자가 데이터 및 노드의 위치를 자유롭게 수정하고 조절할 수 있도록 개발

- 수정할 데이터나 노드의 위치가 많아도 윈도우 GUI를 이용하여 쉽고 빠르게 수정 가능

- 기본적으로 많이 사용하는 그래프 구성 제공

- 테이터의 표현 결과를 미리보기 탭에서 확인할 수 있음

- 사용자는 툴킷 라이브러리에서 제공하는 함수를 활용하여 문서 작성 가능

61-3. GraphViz :

- AT&T 연구소에서 DOT 스크립트 언어를 사용하여 다이어그램을 그리려고 만든 오픈 소스 프로그램, CPL을 사용

- 데이터 기반으로 다이어그램을 그릴 수 있는 명령어 라인 네트워크 그래프 시각화 툴, 일반적인 목적의 흐름도나 트리를 그리는데 적합

- 결과 그래프는 다른 표현 툴의 결과물에 비해 상대적으로 간단하거나 투박

- 결과 그래프를 효과적으로 다듬을 수 있는 다양한 옵션을 함께 제공

- 다른 소프트웨어에서 라이브러리로 사용할 수 있는 기능 제공

- DOT 언어를 사용하여 그래프를 정의

- 정의한 내용을 필터를 이용하여 다양한 이미지 포맷으로 생성

61-4. Processing :

- 기존의 DBN기반으로 만든 프로그래밍 언어

- 디자이너와 아티스트의 프로그래밍 접근도를 높여 주는 소프트웨어

- 프로그래밍이나 계산 과정의 난해함을 최소화하여 디자이너 코딩 몇 줄로 이미지를 만들 수 있도록 고안한 SW

- 가장 많이 알려진 그래픽 프로그래밍 언어

- 관련 있는 웹을 시각화 가능 : 연관 라이브러리, 예시, 관련 문서 등이 많아 사용자가 원하는 정보 획득 쉬움

- 여러 가지 시각화 도구로 구성되어 있는 JavaScript 프레임워크인 Protovis 바라인 그래프, 포스-디렉티드 네트워크 구성과 같은 다양한 기능이 포함

61-5. Fusion Tables :

- 테이블의 데이터를 효과적으로 시각화할 수 있게 도와주는 도구

- 방대한 양의 데이터를 스프레드시트와 같은 테이블에 저장 할 수 있는 온라인 시스템

- 저장된 정보를 사용자 요구에 맞게 처리하고, 그 결과를 시각화 할 수 있음

- MS 엑셀에서 제공하는 차트와 달리

- 테이블을 파일이나 이미지 등 공유 대상으로 지정, 구글 맵, 구글 스프레드시트와 간단히 연동 가능

- 엑셀 CSV 등의 파일 업로드 가능

61-6. Tableau :

- 그래프를 시각화하는 데스크톱 응용 프로그램

- 온라인 지원과 다양한 콘텐츠를 추가 가능

- 시각적인 분석과 리포팅 도구 제공

- 사용 쉬우며 GUI가 직관적

- 사용자의 피벗 테이블이나 크로스 테이블의 작성을 도와주는 도구 주 강력한 기능 제공

- 데스크톱의 인터페이스를 사용하거나 전문적인 출판물에 첨부할 다양한 형태의 그래프를 생성하는 작업을 할 때 사용하기 좋은 프로그램

61-7. TinkerPop :

- 오픈 소스 그래프 소프트웨어인 TinkerPop은 통합된 도구 모음을 생성할 수 있으며, 그래프를 처리하는 LAMP구조와 비슷함

- TinkerPop 은REST 기반의 서버 형태로 시스템을 공개함

- 전문화된 그래프 데이터베이스에 질의 작성하고, 상호 연결되어 있는 일반적인 작업들을 수행할 수 있는 서비스 모음을 제공

61-8. Clustergram :

- Clustergram은 클러스터링과 관련된 표현 소프트웨어

- 계층적 클러스터링을 수행하고, 계통도나 히트 맵 형태로 표현

- 클러스터 개수가 늘어나면 각 데이터 집합 부분에 어떻게 클러스터를 할당하는지 보여 주는 집약 분석 기법

61-9. Spatial information Folw :  
- 특정 정보 안에서 데이터의 흐름을 보여 주는 시각화 기술

- 주로 정보의 흐름을 2차원이나 3차원 공간에 링 형태로 시각화하여 표현

- 일반적으로 정보의 흐름이 많을수록 링의 크기가 커짐

62. 데이터 분석 과정 :

1단계 : 문제 정의 및 계획

- 문제가 명확해야 그 문제를 해결하기 위한 데이터가 어떤 것인지를 추정, 어떤 분석기법을 적용해야 할지 계획

2단계 : 데이터 수집

- 기존 시스템의 데이터베이스, 엑셀파일, 종이 문서, 장비내의 파일, 인터넷 등에서 필요한 자료 수집

3단계 : 데이터 정제 및 전처리

- 수집된 데이터-분석에 사용할 수 없는 경우가 대부분

- 단위의 차이, 결측지, 오류 데이터 등의 보정 필요

- 수집된 데이터를 분석이 가능한 형태로 정돈하는 과정을 데이터 정제 혹은 전처리 과정

4단계 : 데이터 탐색

- 가벼운 데이터 분석

- 전반적인 데이터의 내용을 파악하는 단계

5단계 : 데이터 분석

- 데이터 탐색 단계에서 파악한 정보를 바탕으로 보다 심화된 분석을 수행하는 단계

- 전통적인 통계분석을 포함하여 고급 분석 기법 사용

- 머신러닝 기술 적용

6단계 : 결과 보고

- 데이터의 분석과 해석이 마무리, 내용 정리, 보고

- 결과보고 작성단계에서 데이터 시각화

- 데이터 시각화 : 분석된 결과를 단순 숫자의 나열이 아니라 다양한 그래프나 그림을 통해서 결과를 쉽게 이해할 수 있도록 표현

63. 빅데이터 처리에서 발생하는 부하 종류 :

- 컴퓨팅 부하 : 빅데이터 처리 위한 CPU 등 자원 부하

- 저장 부하 : 빅데이터 처리 과정의 입력 데이터, 중간 데이터, 출력 데이터 등 저장 부하

- 네트워크 부하 : 빅데이터 처리에 사용하는 노드 간 통신 부하

64. 빅데이터 플랫폼에서 해당 부하 제어 예 :

- 컴퓨팅 부하 : CPU 성능 향상 및 빅데이터 응용을 실행하는 클러스터에서의 효과적인 자원 할당

- 저장 부하 : 파일 시스템 개선, 메모리와 파일 시스템의 효과적인 사용, 데이터베이스 기술 향상

- 네트워크 부하 : 노드 간 대역폭을 효과적으로 분배하거나 네트워크에서 가까운 노드를 선택

65. 빅데이터 플랫폼 요소기술과 구조 :

빅데이터 수집 및 정제 :

- 데이터 정제 : 데이터 품질을 향상시키기 위해 데이터의 오류 및 불일치를 감지하고 제거

ETL 프로세스 :

- 데이터 추출 : 데이터 소스에서 데이터 추출

- 데이터 변환 : 데이터 소스에서 추출한 데이터 변환, 균질화 및 정제, 데이터를 해당 데이터웨어하우스 스키마에 적합하도록 변환

> 비즈니스 규칙 및 통합 제약 조건에 따라 데이터웨어하우스에 데이터를 전달하는 필터링 및 확인하는 작업도 수행

- 데이터 로딩 : 변환된 데이터를 데이터웨어하우스로 로딩

빅데이터 처리 :

- 수집 및 정제된 데이터를 효과적으로 분석하는 처리

- 작업을 병렬 및 분산 처리 통해 성능 향상

-병렬 시스템 :

문제 해결을 위해 CPU 등의 자원을 데이터 버스나 지역 통신 시스템 등으로 연결하여 분할된 작업을 동시에 처리하여 계산 속도 빠르게 하는 시스템

- 분산 시스템 :

네트워크에서 분산된 컴퓨터를 단인 시스템 형태로 구동하는 시스템

분산 시스템에 속한 각 노드는 독립된 시스템. 독립 컴퓨터의 집합이나 단일 시스템의 역할을 수행

66. 분산 병렬 컴퓨팅 :

- 다수의 독립된 컴퓨팅 자원을 연결하는 네트워크, 이를 제어하는 미들웨어와 하나의 시스템으로 동작

- 고려요소 3가지 :

1. 전체 작업을 쪼개어 여러 개의 작은 작업으로 나누는 문제

- 워크 플로우처럼 상호 의존적인 작업은 스케줄링등 이용해 적절한 순서로 실행

2. 개별 프로세서에서 계산된 중간 결과물을 프로세서 간에 서로 주고받는 문제

- 효율적인 통신은 성능과 직결

- 단일 시스템은 노드 수만큼 전체 작업을 균등하게 나누고, 이종 시스템은 컴퓨팅 능력에 따라 전체 작업을 나눔, 이 때 노드 간 통신을 최소화하는 기법 고려

> Ex) 그래프 분할 등 반영, 효율적인 자원사용 가능, 성능 향상

3. 서로 다른 프로세서 간의 동기화 문제

- 데이터 병렬 처리에서 동기적 방식의 경우 프로세서는 특정 계산이 끝나거나 특정 데이터를 넘겨 받을 때까지 대기. 즉, 전송자가 수신자에게서 데이터 수신 확인 응답 때까지 대기

- 비동기적 방식 : 결과 메시지를 보낸 즉시 다음 작업을 계속. 이때 프로세서는 기다릴 필요 없음. 계산 과정이 적합한지 확인 필요

67. 맵리듀스 :

- 글 개발. 방대한 양의 데이터를 처리하는 프로그래밍 모델

- 효과적인 병렬 및 분산 처리 제공

- 맵 작업 : 입력 데이터를 처리하여 중간 데이터를 생성

- 리듀스 잡업 : 중간 데이터를 병합하여 출력 데이터 생성

- 맵 작업과 리듀스 작업

>병렬 및 분산 처리, 빅데이터의 신속한 처리 보장

> 맵리듀스 처리 성능 영향 인자 : ㅅㄹ행하는 런타임에서의 입력 데이터 분할, 작업 스케줄링, 노드 고장, 노드 간의 데이터 전송 등

68. 빅데이터 응용) 맵리듀스 함수 호출시 처리 단계 :

- 입력 데이터 읽고 분할하는 단계

- 분할된 입력 데이터를 할당해 맵 작업 수행 후, 중간결과 데이터 통합 분할 단계

- 통합 분할된 중간 데이터를 셔플하는 단계

- 셔플된 중간 데이터를 이용해 리듀스 작업 수행 단계

- 출력 데이터를 생성하고 맵리듀스 처리 종료 단계

69. 빅데이터 분석 :

- 데이터의 효과적인 분석 기술

> 탐구 요인 분석(EFA), 확인 요인 분석(CFA)

>> EFA : 데이터 간 상호관계를 파악하여 데이터를 분석 하는 방법

>> CFA : 관찰된 변수들의 집합 요소의 구조를 파악하기 위한 통계적기반 데이터 분석 방법

- 빅데이터 표현

> 빅데이터 처리 및 분석 결과를 사용자에게 보여 주는 기술

> 처리 및 분석된 결과를 표, 그래프 등을 이용해 쉽게 이용

70. 빅데이터 플랫폼 구조 및 세부 기술 :

- 빅데이터를 통합적으로 다루는 플랫폼

1. 소프트웨어 계층

- 빅데이터 응용 구성, 데이터 처리 및 분석과 이를 위한 데이터 수집 및 정제 수행

> 데이터 처리 및 분석 엔진 : 데이터를 처리하고 분석하는 엔젠

> 데이터 수집 및 정제 모듈 : 빅데이터 분석 엔진을 위한 데이터 수집 및 정제

> 서비스 관리 모듈 : 소프트웨어 계층에서 제공하는 서비스 관리

> 사용자 관리 모듈 : 사용자 관리

> 모니터링 모듈 : 플랫폼 및 인프라스트럭처 서비스 가용성 및 성능 모니터링

> 보안 모듈 : 소프트웨어 계층 보안 관리

2. 플랫폼 계층

- 빅데이터 응용을 실행하기 위한 플랫폼 제공

- 이를 위한 데이터 및 자원 할당을 작업 스케줄링, 데이터 및 자원 할당, 프로파일링, 데이터 관리, 자원 관리 등 통해 수행

> 사용자 요청 파싱 : 사용자 요청 파싱

> 작업 스케줄링 모듈 : 사용자 응용을 실행하는 작업을 스케줄링

> 데이터 및 자원 할당 모듈 : 사용자 응용을 실행하는 데이터 및 자원 할당 모듈

> 프로파일링 모듈 : 데이터 및 자원을 할당하는 자원 및 응용 프로파일링 및 작업 시뮬레이션

> 데이터 관리 모듈 : 사용자 데이터 관리

> 자원 관리 모듈 : 인프라스트럭처 자원 관리

> 서비스 관리 모듈 : 플랫폼 계층에서 제공하는 서비스 관리

> 사용자 관리 모듈 : 사용자 관리

> 모니터링 모듈 : 인프라스트럭처 서비스 가용성 및 성능 모니터링

> 보안 모듈 : 플랫폼 웨어 계층 보안 관리

3. 인프라스트럭처 계층

- 자원 배치, 노드, 스토리지, 네트워크 관리 등 통해 빅데이터 처리 및 분석에 필요한 자원 제공

> 사용자 요청 파싱 : 사용자 요청 파싱

> 자원 배치 모듈 : 사용자에게 제공할 자원 배치

> 노드 관리 모듈 : 인프라스트럭처 내의 노드 관리

> 데이터 관리 모듈 : 인프라스트럭처 내의 스토리지 관리

> 자원 관리 모듈 : 인프라스트럭처 내의 네트워크 관리

> 서비스 관리 모듈 : 인프라스트럭처 계층에서 제공하는 서비스 관리

> 사용자 관리 모듈 : 사용자 관리

> 모니터링 모듈 : 서비스 모니터링

> 보안 모듈 : 인프라스트럭처 계층 보안 관리

71. 가상화 기술 :

- 실제 존재하는 물리적 자원을 추상화하여 논리적 자원 형태로 나타내는 기술

- 기존의 복잡한 물리 장치에서 컴퓨팅 자원을 분리시켜 효율적으로 활용하고 관리

72. 서버 가상화 :

-물리적 서버 하나에 여러 개의 가상 서버를 구축

- 서버 하나에서 각 응용 프로그램과 운영체제가 독립된 환경 보장 및 사용

- 데이터 센터의 가상 머신 Virtual Machine ; VM은 서버 가상화로 동일한 물리적 머신 Physical Machine ; PM을 공유

- PM의 CPU, 메모리, I/O 등 자원 분할하여 사용하므로 서버의 효율 증가

- 데이터 센터의 활용도 : 보통 약 10~15%

- 가상화된 서버의 활용도 : 70%

73. 서버 가상화 분류 :

- 호스트 기반 가상화 :

- 하이퍼바이저가 호스트 운영체제 위에 설치되는 구조

- 장점 : 설치가 쉽고 구성이 편리

- 단점 : 성능 저하

- 베어메탈 기반 가상화 :

- 하이퍼바이저가 호스트 운영체제를 거치지 않고, 시스템 하드웨어와 직접 연결되어 가상 머신을 생성하고 관리

- 장점 : 향상된 성능을 제공하고 실시간 운영체제를 지원할 수 있다는 장점

- 단점 : 운영체제가 위에 없기 때문에 디바이스용 드라이버, 하드웨어 플랫폼 드라이버 등을 포함해야 하고, 설치와 구성이 어려움