빅데이터 솔루션을 활용한 스마트미터 전력량 예측 분석

오진영, 이희철, 최준혁

죽음의 불4조

목차

1. 프로젝트 배경

2016년부터 사람, 사물, 정보가 하나로 연결되는 4차 산업혁명이 시작되면서 스마트 에너지, 스마트 시티, 인공지능, 사물인터넷, 무인자동차, 로봇산업 등 다양한 분야가 메인스트림으로 자리잡으며 필요한 핵심 기술로 빅데이터를 주목 받고 있다. 최근 빅데이터 분야의 이슈 중 이세돌 9단과 알파고와의 대결, 구글 무인자동차의 초당 1GB 규모로 발생하는 센서 데이터를 실시간 분석하며 자율주행에 성공, 미국 대선에서는 실시간 SNS데이터를 활용한 빅데이터 분석만이 트럼프 의 승리를 예측하는 등 빅데이터 분석은 각 분야에서 큰 발전을 도모하고 있다.

게다가, 최근 인공지능이 컴퓨터 기술의 발달로 여러 분야에서 성과를 거두고 있고, 이는 과거의 축적된 데이터를 좋은 성능의 컴퓨터로 머신러닝을 적용하여 빠르게 처리가 가능한 기술력이 바탕이 됐기 때문이다. 최근 2년 동안 발생한 데이터는 전 세계 데이터의 80%를 차지하며, 향후 지구상에서 발생하는 데이터의 양은 2020년까지 약 35,000엑사바이트 수준까지 증가할 것이라고 예측하고 있다.

출처 : 서울경제>경제동향

과거에서 현재까지 데이터를 분석하고 의미를 도출해 내 미래를 예측하는 빅데이터 분석은 데이터를 단순히 크기만으로 판단하는 것이 아닌 데이터의 형식, 처리속도, 의미 등을 내포한 개념으로 단순히 통계적 분석이 아닌 데이터의 수집, 저장, 검색, 분석 등의 모든 과정을 총칭한다. 빅데이터의 핵심기술에는 하둡 소프트웨어가 중심에 있다. 하둡은 대용량의 자료를 처리할 수 있는 큰 컴퓨터 클러스터에서 동작하는 분산 응용 프로그램을 지원하는 프리웨어 자바 소프트웨어 프레임워크이며 이는 분산처리 시스템인 구글 파일시스템을 대체할 수 있는 하둡 분산 파일시스템(HDFS: Hadoop Distributed File System)과 맵리듀스[1]를 구현한 핵심 기술이다. 하둡은 HDFS와 맵리듀스의 코어 프로젝트를 중심으로 주변에 서브프로젝트들이 진행되면서 하둡 에코시스템이 구성되고, 이는 시스템을 하나의 생태계로 지칭한다

본 파일럿 프로젝트는 빅데이터 처리과정을 4개의 레이어(수집, 적재, 탐색/처리, 분석/응용)로 나눠 각 역할에 맞는 프로젝트를 선정 및 구현하여 하둡에코시스템을 활용하여 빅데이터 솔루션을 구축하고 운용해보자 한다.

[1] 맵리듀스(MapReduce)는 구글에서 대용량 데이터 처리를 분산 병렬 컴퓨팅에서 처리하기 위한 목적으로 제작하여 2004년 발표한 소프트웨어 프레임워크다.

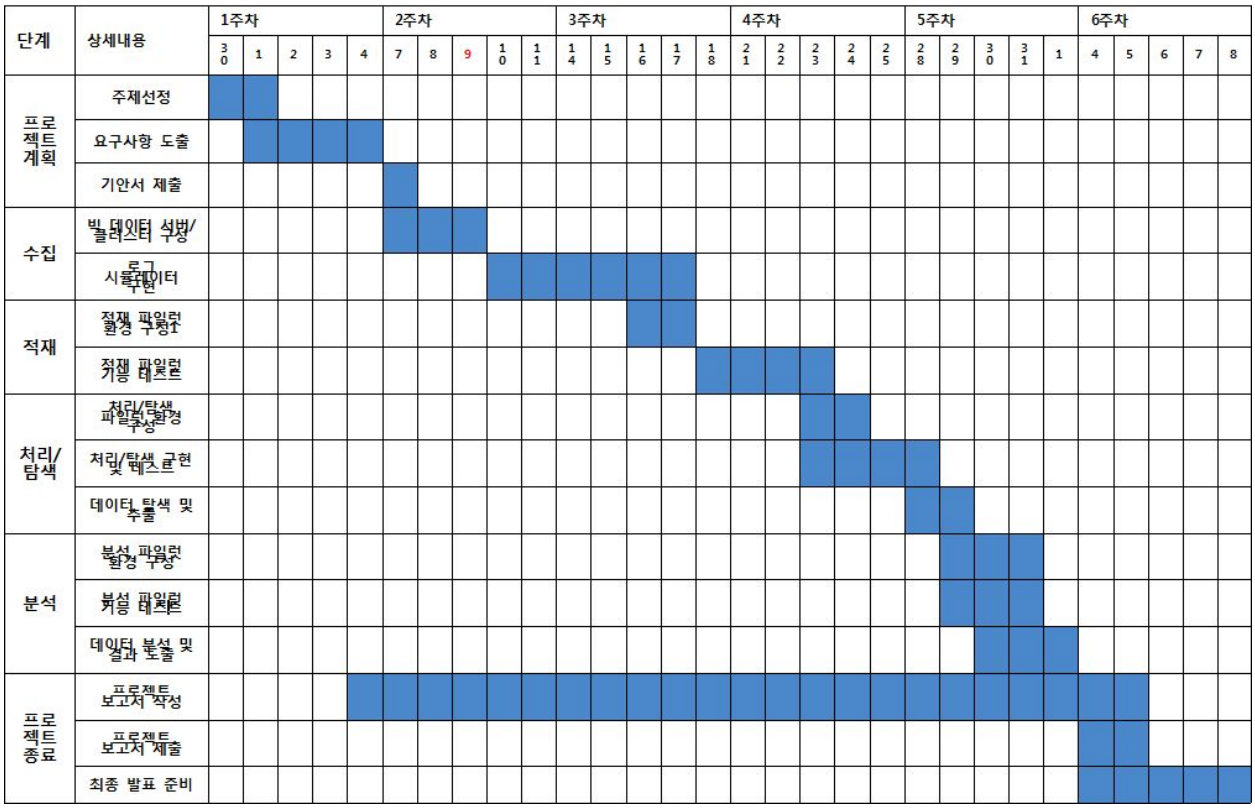
2. 목표 및 계획

2.1 프로젝트 목표

본 프로젝트에서 구축하고자 하는 빅데이터 솔루션은 실시간 대용량 데이터를 수집되고 있는 가상의 환경을 세팅하여 수집단위별로 용량은 작지만 일정 주기를 기준으로 발생되는 대량의 데이터를 수용할 수 있는 환경을 구축- 운용 하는것을 목표로 시작한다.

이는 파일럿 프로젝트로써 빅데이터 솔루션 성공적 시험운영에 목적을 두고 시스템에 적절한 소프트웨어/하드웨어 아키텍처 구축환경을 조성하는 것에 큰 역할이 부여된다. 수집되는 데이터의 특징에 따라 소프트웨어 장/단점을 파악하여 목적에 부합한 서브 프로젝트의 선정이 필요하며, 아키텍처 구성을 위해 6주의 프로젝트 기간에 맞춰 일정을 계획하고 구축 환경에 중점을 두고 프로젝트를 진행했다.

2.2 프로젝트 계획



3. 프로젝트 도메인

3.1 스마트 그리드 소개

현재의 전력시스템 상에서는 우리가 실제로 사용하는 전기보다 15% 정도 많이 생산하도록 설계되어 있다. 이는 전력의 최대소비량에 맞춰진 양으로 혹시라도 더 많이 사용할 경우에 대비해 전기를 미리 확보해 놓는 것이다. 꼭 필요한 만큼 전기를 생산하거나 생산량에 맞춰 전기를 사용할 수 있다면 전기를 더 효율적으로 사용하면서 지구온난화도 막을 수 있다는 생각으로 시작된 사업으로 스마트그리드는 전력망에 정보통신기술을 융합해 전기사용량과 공급량, 전력선의 상태까지 알 수 있는 기술로 에너지 효율성을 극대화할 수 있다.

스마트그리드의 핵심은 전력망에 직비(Zigbee)[2], 전력선 통신 등의 정보통신기술을 합쳐 소비자와 전력회사가 실시간으로 정보를 주고받는 것에 있다. 따라서 소비자는 전기요금이 쌀 때 전기를 쓰고, 전자제품이 자동으로 전기요금이 싼 시간대에 작동하게 하는 것도 가능하다.

출처 : (재)한국스마트그리드사업단 홈페이지

- 전력망(Grid): + 정보통신(Smart) = 스마트 그리드
- 스마트 미터 = 가정/주택에 장착된 스마트미터는 전력사용량을 모니터링 하고 전력 정보를 실시간으로 만든다.
- 다양한 분야의 전력망 중 가정/주택용 전력 사용 데이터만 사용하며 100가구의 전력정보를 기준으로 진행하는 파일럿 프로젝트이다.
- 본 빅데이터 솔루션은 이 데이터들을 수집→ 적재 →처리(탐색) → 분석(응용) 를 통하여 사용자(가정용)에게 편의성과 안전성을 지원하는 서비스 이다.

[2] **직비**(ZigBee)는 소형, 저전력 디지털 라디오를 이용해 개인 통신망을 구성하여 통신하기 위한 표준 기술이다(위키백과)

3.2 스마트미터 가구별 소비 전력 데이터

- 100세대의 스마트미터로부터 발생하는 전력사용 로그를 수집해 전력 사용량 분석
- 실시간(초) 소비 전력 데이터는 일정 구간 이상치를 기록하는 데이터를 필터링하여 해당 고객에게 정보를 제공해주는 서비스를 위한 데이터셋
- 15분 누적 소비 전력 데이터는 스마트미터가 15분 단위로 소비 전력량을 수집하여 솔루션에 저장하고 이후 데이터 분석을 위한 데이터셋

요구사항	각 세대 스마트미터 발생 로그 파일 수집 후 전력사용량 상태 점검(실시간)	
데이터의 종류	1인~9인 가구 100세대 전력량	
데이터 발생 주기	1초	
데이터 수집 주기	실시간	
데이터 수집 규모	100세대/초 (1일 수집규모 : 약 1GB)	
데이터 타입	텍스트(UTF-8), 반정형	
데이터 분석 주기	분/시간/일/주	
데이터 처리 유형	실시간	
데이터 구분자	콤마(,)	
데이터 스키마	발생 일시	20191014045403(2019년 10월 14일 4시 54분 03초)
	고객 번호	H000~H100
	전력(kw)	실시간 전력사용량/1초
	mac address	00:00:00:00
예시	20190101000001, H001, 0.001162, a1:b1:c1:d1 정보 : 2019년 1월 1일 00시 00분 1초에 고객정보 H001의 전력량 0.001162발생 고유 주소는 a1:b1:c1:d1	

요구사항	각 세대 스마트미터 발생 로그 파일 수집 후 전력사용량 상태 점검(15분)	
데이터의 종류	1인~9인 가구 100세대 전력량	
데이터 발생 주기	15분	
데이터 수집 주기	15분	
데이터 수집 규모	100세대	
데이터 타입	텍스트(UTF-8), 반정형	
데이터 분석 주기	시간/일/주/월/년	
데이터 처리 유형	배치	
데이터 구분자	콤마(,)	
데이터 스키마	발생 일시	201901011215(2019년 1월 1일 12시 15분)
	고객 번호	H000~H100
	전력(kw)	전력사용량/15분
	mac address	00:00:00:00
예시	20190101000001, H001, 0.1425, a1:b1:c1:d1 정보 : 2019년 1월 1일 00시 00분 1초에 고객정보 H001의 전력량 0.1425발생 고유 주소는 a1:b1:c1:d1	

가구별 스마트미터 수집 데이터 스키마(좌: 1초 주기 / 우: 15분 주기)

3.3 가구별 고객정보 도메인

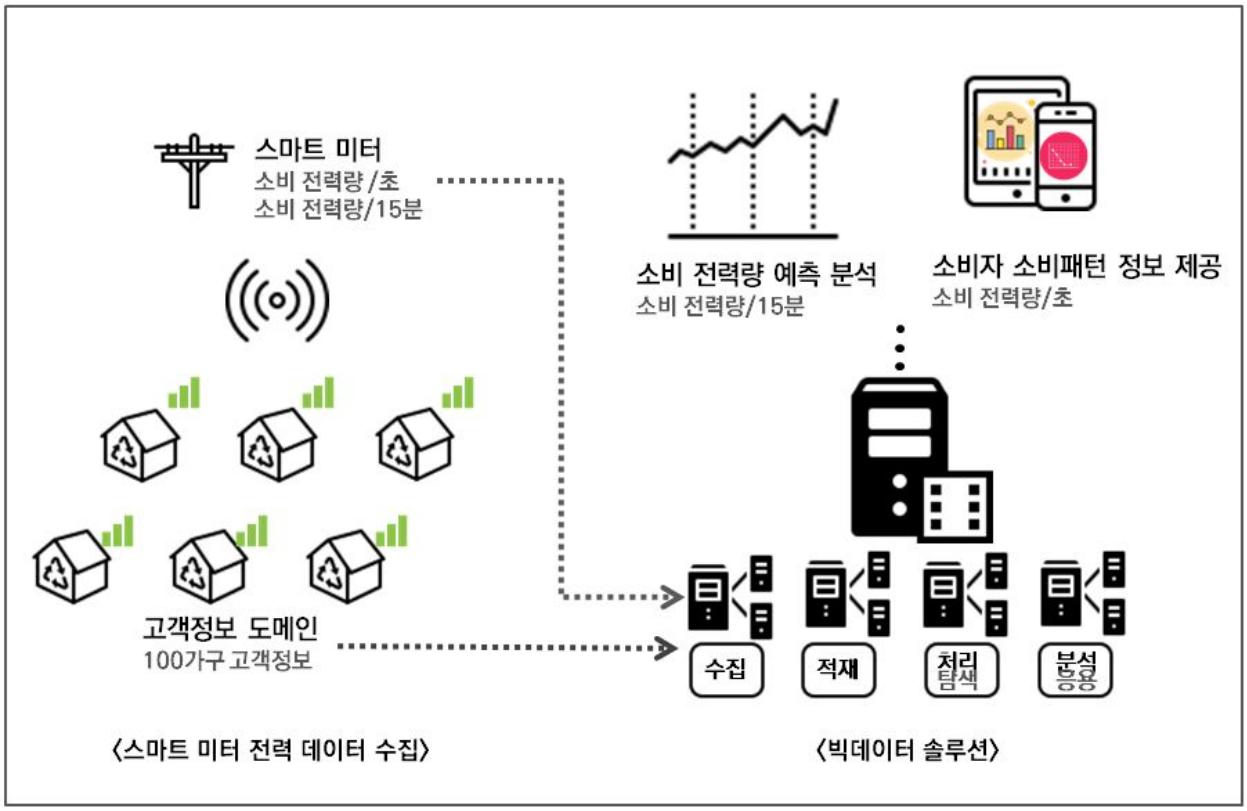
- 100세대의 고객정보 도메인을 스마트 미터에서 수집되는 각 세대당 전력데이터를 해당 도메인 정보를 기반으로 가중치 적용을 위해 사용되는 데이터셋

요구사항	고객정보 도메인	
데이터 규모	100세대	
데이터 타입	csv	
데이터 처리 유형	배치	
도메인 정보	고객번호	H000~H100
	가구 구성원 수	0 ~ 9인
	mac address	00:00:00:00

user_id	family	mac
H001	2	f2:42:2d:66:8
H002	4	90:09:94:24:f
H003	3	60:36:5c:1e:d
H004	2	ac:59:52:b5:5
H005	5	82:f5:7c:8a:4
⋮		
H095	3	42:c5:aa:b7:9
H096	4	82:23:b6:67:6
H097	3	2a:0c:c8:cf:31

가구별 고객정보

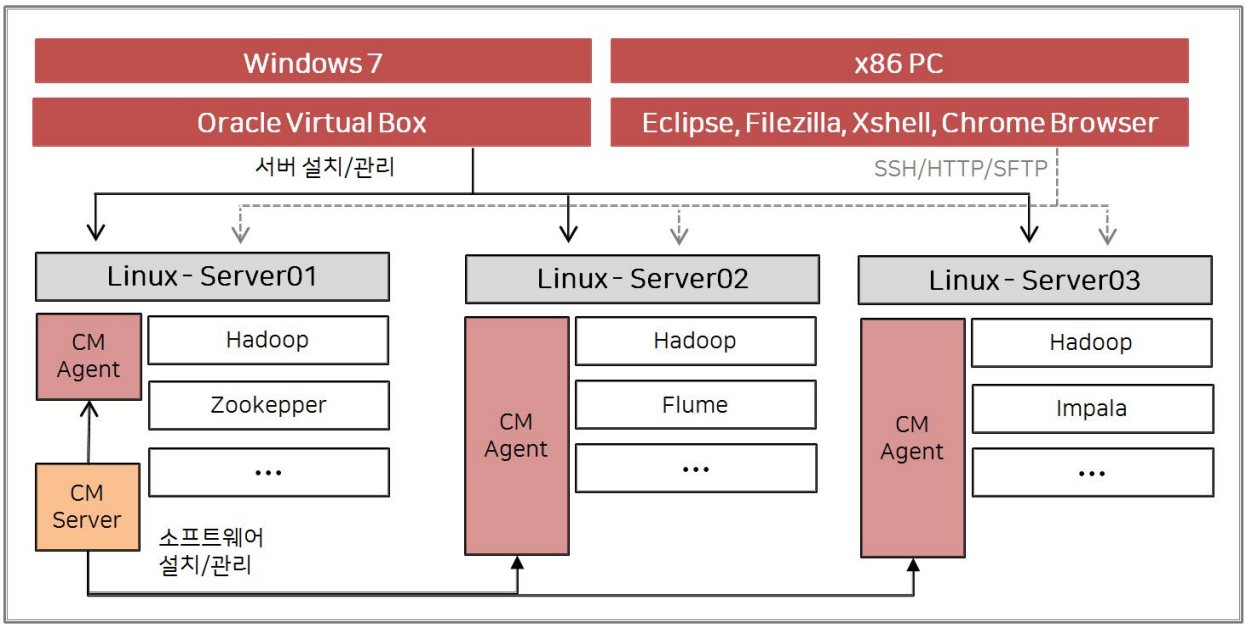
3.4 프로젝트 도메인 구성



프로젝트 도메인 흐름

- 실제 100가구의 스마트 미터를 운영할 수 없기때문에 스마트 미터 로그 시뮬레이터 작성한다.
- 스마트 미터가 수집하는 데이터는 전력생성 로그 시뮬레이터로 한국전력공사의 가구수 / 월별 전력사용량 통계데이터를 기반으로 데이터를 수집 주기에 맞춰 소비 전력량을 생성한다.
- 스마트 미터 데이터는 수집 → 적재 → 처리/탐색 → 분석/응용 과정을 거치고 각 단계별로 활용하기 쉬운 데이터셋으로 재구성한다.
- 탐색과분석, 머신러닝 기법을 적용해 전력사용량 예측 등 데이터 마이닝 작업을 진행한다.

4. 시스템 아키텍처



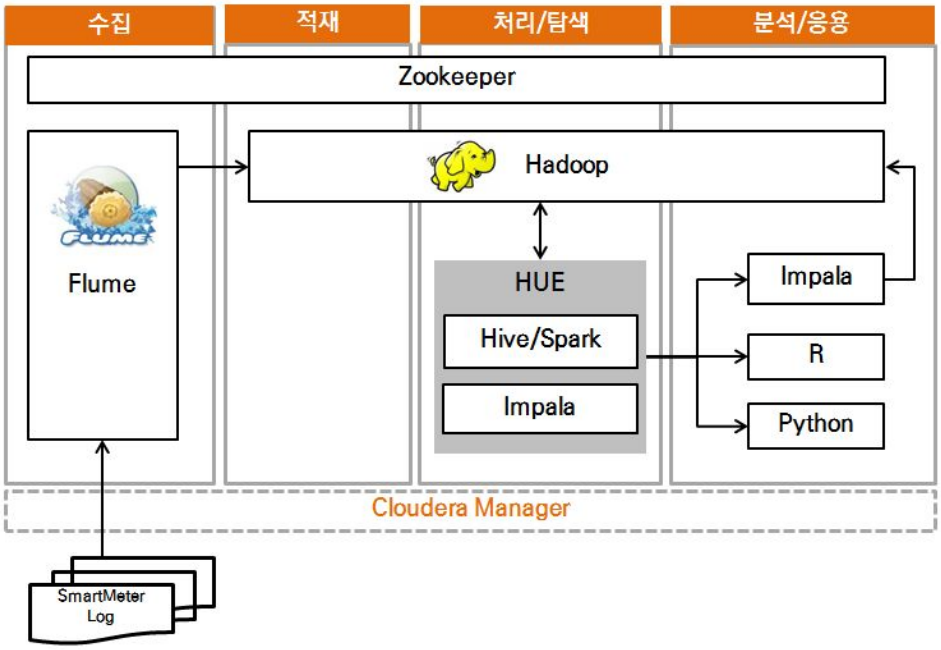
하둡 에코시스템을 운용하기 위해서는 가상 머신 환경이 필요하다. 가상 머신 환경을 구축하기 위해서 오라클 버추얼박스를 설치했고, 이를 통해 3대의 가상머신에 CentOS 리눅스를 설치했다. 이 3대의 리눅스에서 빅데이터 에코시스템을 설치해 파일럿 환경을 단계적으로 구축했다.

가상 머신 위에 설치된 CentOS 리눅스 서버에는 설치 및 구성을 자동화하는 CM(Cloudera Manager)을 공통을 설치 했으며 CM으로 모든 소프트웨어들을 3대의 가상 머신에 설치/관리하고, 설치된 소프트웨어들의 주요 리소스와 기능들을 모니터링하고 관리하는 부분을 중점적으로 활용했다.

Eclipse는 1초당 그리고 15분당 전력사용량을 발생하는 로그 시뮬레이터 어플리케이션을 개발하는데 사용했다. Filezilla는 FTP 프로그램으로서 리눅스 원격서버의 디렉토리 관리하는데 사용했고, Xshell은 SSH 접속을 쉽고 간편하게 하기 위해 사용했다. 마지막으로 CM의 특정상 브라우저는 Chrome Browser 만 사용했다.

5. 프로젝트 아키텍처

실시간 대용량 데이터를 수집하는 대상으로 빅데이터 프로젝트를 진행한다면 전력사용 가정용 가구에 맞춰 수십~수백 대의 하둡 클러스터 노드를 구성해야 하지만 그러한 빅데이터 환경을 구성하는 것은 본 프로젝트에서 현실적으로 불가능하다. 따라서 프로젝트에 적용가능한 개인PC를 활용할 수 있는 수준으로 소규모 빅데이터 파일럿 환경을 구성하고 파일럿 환경에서도 빅데이터의 핵심 기술과 기능들을 모두 다룰 수 있게 아래 그림과 같이 하둡 에코시스템 구축을 목표로 했다.

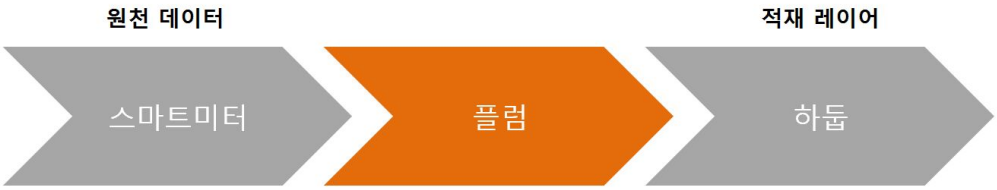


각 오픈소스 프로젝트는 크게 수집, 적재, 처리/ 탐색, 분석/응용 레이어로 분류하여 3대의 가상머신을 만들고, 총 8개의 소프트웨어를 설치하여 솔루션 아키텍처를 구성했다.

프로젝트만을 위한 개인용 파일럿 환경이지만 빅데이터의 모든 기술 요소를 다 갖춘 환경으로서 이를 수작업으로 설치 및 구성하기는 전문가들도 하기 어려운 작업이기 때문에 본 파일럿 프로젝트에서는 빅데이터 자동화 관리툴인 클라우데라의 CM(Cloudera Manager)을 활용했다. 클라우데라는 하둡을 포함한 에코시스템을 편리하게 설치 및 관리해 주는 역할을 한다. 6장부터 각 레이어들의 역할과 레이어에 배치돼 있는 주요 소프트웨어들의 기능을 구체적으로 설명하도록 다루었다.

5.1. 수집 레이어

빅데이터 시스템 구축은 수집에서부터 시작된다. 빅데이터 프로젝트에서는 여러 공정 단계가 있는데, 그중 수집이 전체 공정 과정의 절반 이상을 차지한다. 빅데이터 수집은 일반적인 수집과 달리 수집 영역이 조직내의 전체 시스템에서부터 외부 시스템(SNS, 포털, 정부기관)에 이르기 까지 매우 광범위하고 다양하다. 프로젝트 초기에는 이러한 수집 대상 시스템을 선정하고, 그에 따른 연동 규약을 협의 및 분석하는데 엄청난 리소스가 투입된다. 또한 수집 실행 단계에선 업무요건과 환경의 변화로, 이전 단계에서인 수집 계획 수립으로 다시 돌아가는 경우가 빈번하게 발생하며, 그로 인한 계획과 실행 단계가 여러 차례 반복돼 가며, 수집 인터페이스를 수정하는 어려움이 있다.



5.1.1 플럼

본 프로젝트에서는 원천 데이터인 로그 시뮬레이터에서 1초 그리고 15분마다 전송되는 전력량, 즉 로그파일을 수집하기 위해 플럼을 사용했다. 플럼(Flume)은 빅데이터를 수집할 때 다양한 수집 요구사항들을 해결하기 위한 기능으로 구성된 소프트웨어다. 데이터를 원천으로부터 수집할 때 통신 프로토콜, 메시지 포맷, 발생 주기, 데이터 크기 등으로 많은 고민을 하게 되는데 플럼은 이러한 고민을 쉽게 해결할 수 있는 기능과 아키텍처를 제공한다.

플럼 메커니즘은 Source, Channel, Sink 만을 활용하는 매우 직관적인 구조를 갖는다. 플럼의 Source는 데이터를 로드하고, Channel 에서 데이터를 임시 저장해 놓았다가, Sink를 통해 목적지에 데이터를 최종 적재한다. 스마트미터 100대에서 생성되는 정보를 수집하는 로그 파일이 로그 시뮬레이터에 의해 만들어지는데, 플럼 메커니즘을 통해 발생과 동시에 플럼 에이전트가 수집해서 하둡에 전송하는 기능을 한다.

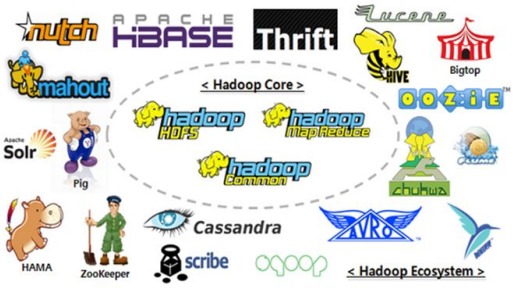
5.2 적재 레이어

이번장에서는 수집한 데이터를 어디에, 어떻게 저장할 것인가를 다룬다. 수집한 데이터는 데이터의 성격에 따라 처리 방식과 적재 위치가 달라질 수 있다. 크게는 데이터 발생주기에 따라 일괄 배치성 데이터인지, 실시간 스트림 데이터인지를 판단해야 하고, 데이터의 형식에 따라 가공처리나 사전 검증 작업을 할 것인지도 판단해야 한다. 적재한 데이터를 어떤 비지니스 요건에서 활용하느냐에 따라 적재 대상 위치가 달라질 수도 있는데, 이는 데이터 적재 후 데이터 분석 방식과 데이터 활용성 및 연관된 업무 시스템의 성격에 따라 적재 저장소가 달리 구성돼야 함을 의미한다. 본 프로젝트에서는 스마트 미터의 대용량 로그 파일을 적재하는 배치성 처리로 구성했다.

본 프로젝트에서 하둡의 활용방안은 스마트 미터에서 발생하는 로그가 큰 파일이기 때문에 HDFS의 특정 디렉터리에 일자 단위로 파티션해서 적재한다. 이렇게 일 단위로 장기간 적재된 데이터는 일/주/월/년별로 스마트 미터의 다양한 시계열 집계 분석을 할 수 있다. 이때 임팔라가 활용되고, 분산 병렬 처리 작업을 위해 맵리듀스 프로세스가 내부적으로 작동한다. 임팔라로 분석된 결과는 다시 HDFS의 특정 영역(Hive Data Warehouse)에 저장되고, 이 데이터가 스마트 미터의 고급 분석으로 까지 확장해서 사용된다.

5.2.1 하둡

하둡은 빅데이터의 핵심 소프트웨어이다. 빅데이터의 에코시스템들은 대부분 하둡을 위해 존재하고 하둡에 의존해서 발전해 가고 있다. 하둡은 크게 두 가지 기능이 있는데, 첫 번째가 대용량 데이터를 분산 저장하는 것이고, 두 번째는 분산 저장된 대용량 데이터를 분석하는 기능이다. 이 가운데 대용량 데이터 처리를 위해 분산 병렬 처리 기술을 사용한다.



<출처: 아파치 소프트웨어 재단>

분산 병렬 처리 기술에서 병렬 처리라는 의미는 원래 CPU는 한 순간의 하나의 일 밖에 못한다. 마치 컴퓨터에서 수많은 프로그램이 동시에 되는 거 같고, 워드를 치면서 영화도 잘 나오고, 심지어 한쪽에서 게임도 할 수 있지만, 이 모든 것이 한순간에는 하나의 일만한다. 물론 얼마 전부터 나온 멀티 코어 때문에 실제로 몇 가지를 동시에 할 수 있다. 이때 몇 가지의 기준은 코어 수가 될 수 있다. 결국 CPU는 한번에 하나의 명령어만 처리한다. 그것을 시분할로 처리 하게 된다. 시분할이라는 것은 일들을 줄을 세워서

한가지씩 일들을 처리함을 의미한다. 이에 대해서 병렬 처리라고 하면 여러가지를 동시에 처리한다는 의미이다.

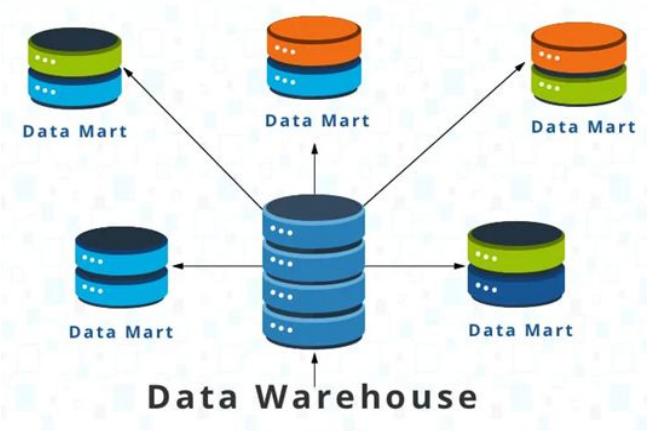
분산처리 컴퓨팅은 한가지 일을 여러개로 나눠서 다른 컴퓨터에 보내 처리를 하게 하고 결과를 모으는 형태이다. 분산 컴퓨팅은 과거 한대의 컴퓨터의 처리 능력이 부족할 때 대용량의 연산처리를 위해서 사용했고,, 방대한 양의 정보를 분석하는 것을 이렇게 분산하여 처리하기도 했다.

5.2.2 주키퍼

본 프로젝트에서 주키퍼는 직접적으로 사용하진 않지만 하둡의 분산 노드 관리에 사용한다. 수십~수천 대의 서버에 설치돼 있는 빅데이터 분산 환경을 더욱 효율적으로 관리하기 위해서는 서버 간의 정보를 쉽고 안전하게 공유해야 한다. 공유된 정보를 이용해 서버 간의 중요한 이벤트(분산락, 순서제어, 부하 분산, 네임서비스 등)를 관리하면서 상호작용을 조율해 주는 코디네이터 시스템이 필요한데, 이것이 바로 분산 코디네이터인 아파치 주키퍼(Apache Zookeeper)다. 이 주키퍼를 활용해서 데이터가 유실되지 않게 클러스터 멤버십 기능 및 환경설정의 동기화 등을 위해 사용한다.

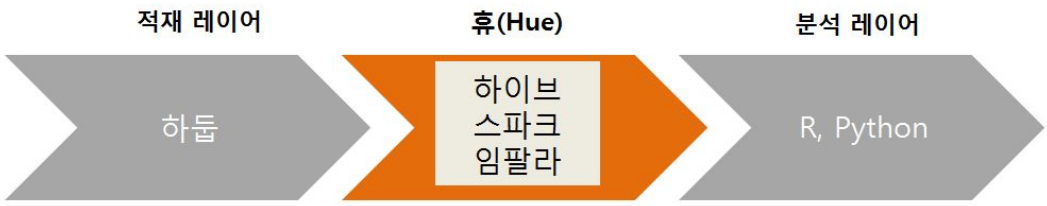
5.3. 처리/탐색 레이어

빅데이터 처리 및 탐색 영역은 적재된 데이터를 가공하고 이해하는 단계다. 특히 데이터를 이해하는 과정에서는 데이터들의 패턴, 관계, 트렌드 등을 찾게 되는데, 이를 탐색적 분석(EDA, Exploratory Data Analysis)이라고도 한다. 탐색 과정은 분석에 들어가기에 앞서 빅데이터의 품질과 인사이트를 확보하는 매우 중요한 단계이다. 덩치 큰 비정형 데이터를 정교한 후처리 작업(필터링, 클린징, 통합, 분리 등)으로 정형화해서 데이터의 직관성을 확보하고, 업무 도메인에 대한 이해를 바탕으로 충분한 탐색적 분석을 진행했을 때 빅데이터를 통한 미래의 통찰력과 비즈니스 가치의 창출이 가능해진다. 그리고 탐색 결과는 곧바로 분석 마트를 위한 기초 데이터로 활용되며, 이러한 일련의 처리/탐색 과정을 거쳐 빅데이터 웨어하우스(Warehouse)가 만들어 진다.



출처 : 위키백과

데이터 웨어하우스 안에서는 다양한 QL을 사용할 수 있고 데이터를 처리/탐색 과정을 거치면서 빅데이터 분석용 마트가 최종적으로 만들어진다. 이때 빅데이터 마트 모델을 통합, 요약, 집계 등을 리포팅하는 현황 분석 모형으로 만들 수 있고, 이를 넘어서 데이터의 패턴과 트렌드를 분석해 미래를 예측하는 고급 분석 모형으로 만들 수도 있다.



본 프로젝트에서는 빅데이터 웨어하우스를 만들기 위한 다양한 처리/탐색 도구들이 있지만 성능이 우수한 임팔라를 사용했다. 하둡에 적재된 방대한 전력사용량 데이터를 임팔라를 이용해서 가구원수별, 월별, 시계열별로 데이터를 정제하여 분석할 때 용이하도록 데이터셋을 만들었다.

5.3.1 임팔라(Impala)

여기 처리/탐색 레이어에서는 임팔라 쿼리를 사용하여 정제/변형/통합/분리/탐색 등의 작업을 수행하고, 데이터를 정형화된 구조로 정규화해 데이터 마트를 만든다. 임팔라의 특징은 하이브 쿼리언어를 사용하지만 속도면에서 3~4배이상 빠른 응답 속도를 보여준다. 그 이유는 하이브는 자바로 만들어졌지만 임팔라는 C++ 기반으로 만들어졌으며, 별도의 실행엔진을 사용하므로 맵리듀스 프로그래밍을 할 필요가 없기 때문이다.

5.3.2 휴(Hue)

빅데이터 탐색/처리는 장기간의 반복 작업이면서 그 과정에 있어 많은 도구들이 활용된다. 주로 하둡을 기반으로 하이브, 피그, 우지, 스쿱, 스파크 등이 해당되며 이를 접해보지 못한 일반 분석가 또는 업무 담당자들이 직접 사용하기에는 어려움이 있다. 빅데이터 기술이 성숙해지면서 이러한 기술의 복잡도를 숨기고 접근성과 편의성을 높은 소프트웨어들이 만들어졌는데, 그중 하나가 클라우데라에서 만든 휴(Hue)이다. 휴는 다양한 하둡의 에코시스템의 기능들을 웹UI로 통합 제공하여 사용자들에게 편의성을 높여주었다. 본 프로젝트에서는 휴를 통해 스파크, 임팔라를 사용하였다.

5.4 분석/응용 레이어

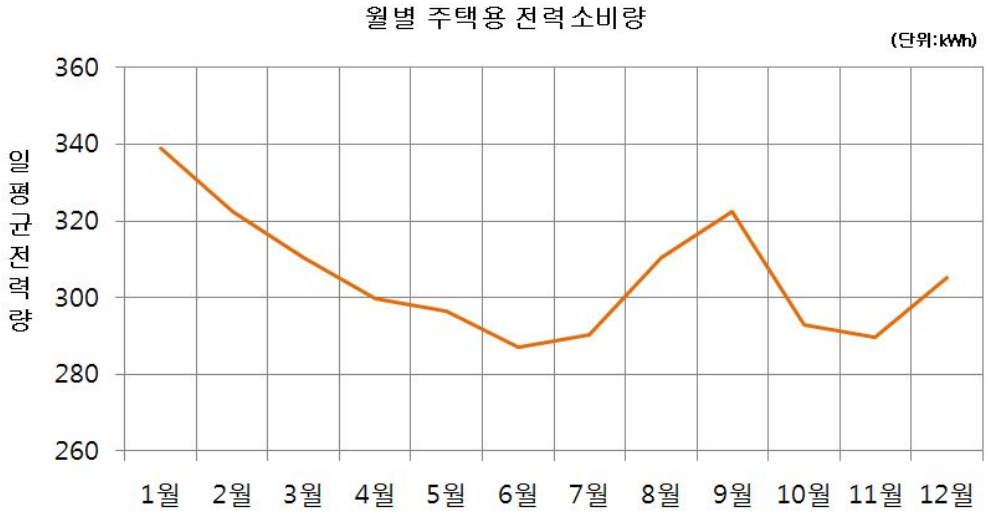
이번 장에서는 2014년부터 2018년까지의 스마트 미터에서 15분 마다 발생한 로그 데이터들이 저장되어 있는 하둡파일시스템(HDFS)에서 임팔라로 만든 데이터 마트를 이용하여 전체 가구 전력소비 패턴을 분석했다.

5.4.1 주택용 전력소비량 현황

2014-2018 연도별 월평균 전력소비량

(단위:kWh)

가구 구성원수	1월	2월	3월	4월	5월	6월	7월	8월	9월	10월	11월	12월	평균	최소-최대 차이
2014	339	323	310	300	297	287	290	310	322	293	289	305	305	52
2015	352	335	322	311	308	298	302	322	335	304	301	317	317	54
2016	353	336	323	312	309	299	302	323	336	305	301	318	318	54
2017	338	322	309	299	296	286	290	309	321	292	289	304	305	52
2018	342	325	313	302	299	289	293	313	325	295	292	307	308	52
평균	345	328	315	305	302	292	295	316	328	298	294	310	311	53



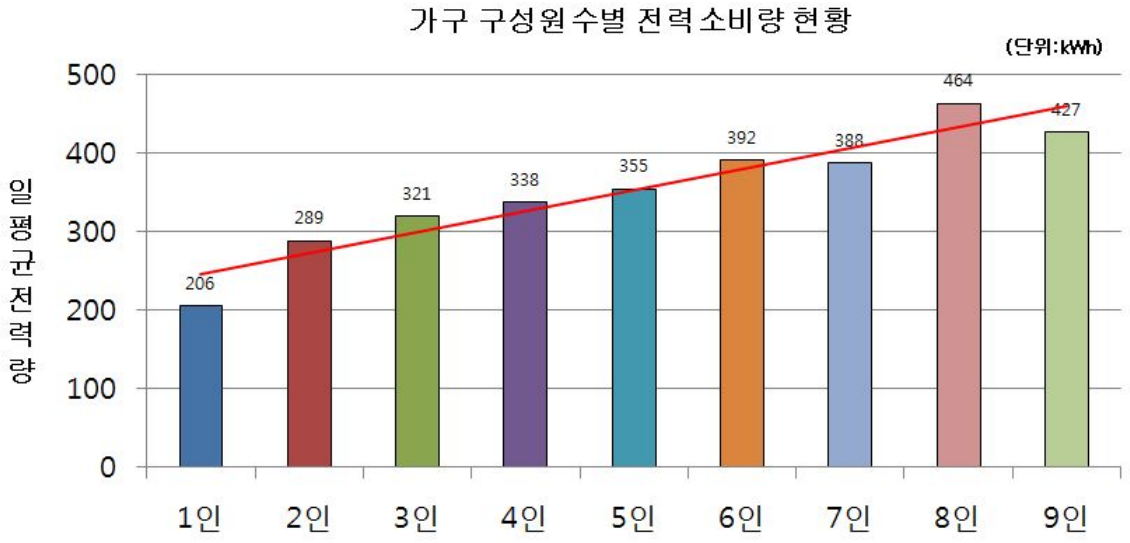
위 표는 스마트 미터에서 발생한 로그 데이터 기반으로 도출한 월별 전력소비량 추이이고, 그래프는 2014-2018 월평균 전력소비량을 나타낸 것이다. 표와 그래프에 의하면, 주택용 전력수요는 동절기 및 하절기에 해당되는 월에서 높은 경향을 보였고, 2014-2018년 모두 동절기에 더 많은 전력을 소비하는 것으로 나타났다. 연도별로 최대 전력소비량 값과 최소 소비량의 차이를 보면 2014년의 경우 52kWh, 2015년 54kWh, 2016년 54kWh, 2017년 52kWh, 2018년 53kWh로 연도별 월간 최대 전력소비와 최소

전력 소비 격차는 50kWh 내외이다. 또한 1월 평균전력량이 339kWh로 가장 높고 6월 287kWh로 낮아졌다가 다시 9월에 322kWh로 높아지는 것으로 나타났다. 이처럼 동절기와 하절기가 평균전력소비량이 높은 것으로 나타났다.

2014-2018 가구 구성원수별 월평균 전력소비량

(단위:kWh)

가구 구성원수	1월	2월	3월	4월	5월	6월	7월	8월	9월	10월	11월	12월	평균	최소-최대 차이
1	229	218	210	202	200	194	196	210	218	198	196	206	206	35
2	321	305	294	284	281	272	275	294	305	277	274	289	289	49
3	356	339	327	315	312	302	306	326	339	308	305	321	321	55
4	375	357	344	332	328	318	322	344	357	324	321	338	338	58
5	394	375	361	349	345	334	338	361	375	341	337	355	355	60
6	436	414	399	385	381	369	373	398	414	376	372	392	392	67
7	431	410	395	381	377	365	369	395	410	373	368	388	388	66
8	515	491	472	456	451	437	442	471	490	446	440	464	465	79
9	475	451	433	419	414	401	405	433	451	410	404	426	427	75
소계/ 평균	393	373	359	347	343	332	336	359	373	339	335	353	354	60



다음 위 표는 2014-2018 가구 구성원수별 월평균 전력 소비량을 비교한 것이고, 그래프는 2014-2018 가구원수별 평균 전력소비량을 나타낸 것이다. 표와 그래프에 의하면, 가구 구성원수가 많을수록 전력소비량도 증가하고 있어, 가구 구성원과 전력소비는 양(+)의 상관관계가 강하게 나타날 수 있는 가능성이 크다. 또한 가구 구성원별 최대 전력소비량 값과 최소 소비량의 차이를 보면 1인 가구의 경우

35kWh로 가장 적었고, 8인 가구일 때 79kWh로 가장 높게 나와 가구 구성원수가 많을수록 최대 전력소비와 최소 전력 소비의 격차가 큰것으로 나타났다.

따라서 주택용 소비전력량 현황은 동절기 및 하절기에 주택용 소비전력량이 크게 늘어나는 것으로 나타났으며, 가구 구성원수가 많을수록 소비전력량이 늘어나는 양(+)의 상관관계 나타나고 있는 것으로 보여진다.

5.4.2 일일 전력소비량 예측

5.4.2.1 LSTM

2014-2018 전력데이터를 이용하여 일별 전력수요량을 예측했다. 또한 기상자료개방포털를 이용하여 2014-2018 기상데이터(평균기온, 최저기온, 최고기온, 평균 이슬점온도, 평균 현지기압, 평균 해면기압, 합계 일사량, 평균 지면온도)를 전력수요량을 예측하는데 변수로 사용하였다. 다음은 전력수요량을 예측하는데 사용할 dataset이다.

DATASET

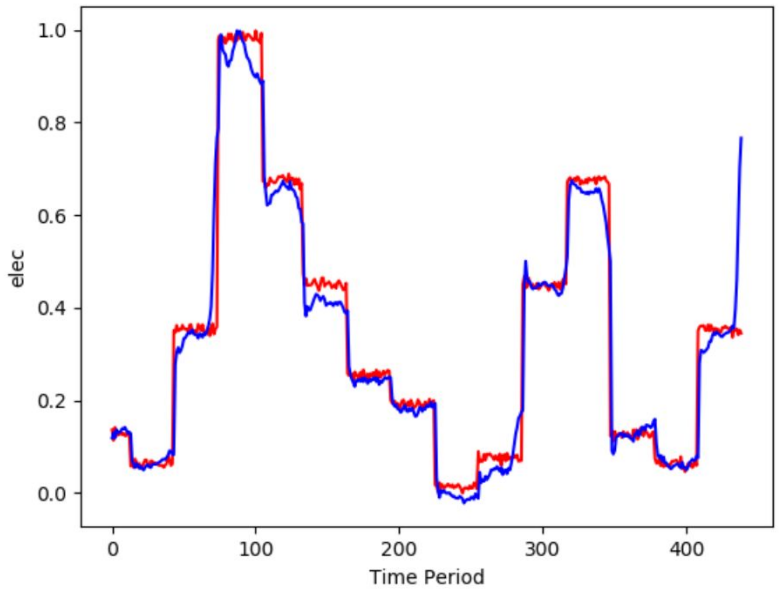
date	평균기온(℃)	최저기온(℃)	최고기온(℃)	평균 이슬점온도(℃)	평균 현지기압(hPa)	평균 해면기압(hPa)	합계 일사량(MJ/m2)	평균 지면온도(℃)	elec
20140101	5.1	0.4	9.7	-2.4	1007.4	1015.9	9.79	1.8	847701.4256
20140102	2.6	-2.2	9.1	-6.1	1013.1	1021.7	11.01	1.3	847868.9795
20140103	2.1	-3.4	9.6	-5.5	1009.3	1018	8.42	0.7	848362.2574
20140104	1	-2.7	6.5	-6.2	1011	1019.7	11.58	0.6	847253.1833
20140105	-0.8	-5.9	5.5	-6.3	1015.5	1024.3	10.97	-1.3	848006.2568

데이터셋은 2014년부터 2018년까지의 일별 데이터로 총 1826개 행이 있고 하이퍼 파라미터는 다음과 같다.

<하이퍼 파라미터>

Input_data_column_cnt : 9	입력데이터의 컬럼 개수
output_data_columns_cnt : 1	결과데이터의 컬럼 개수
seq_length : 365	1개 시퀀스 길이
rnn_cell_hidden_dim : 20	각 셀의(hidden)출력 크기
forget_bias : 1.0	망각편향(기본값:1.0)
num_stacked_layers : 1	stacked LSTM layers 개수
keep_prob : 1	dropout 할 때 keep 할 비율
epoch_num : 1000	에폭 횟수
learning_rate : 0.01	학습률

LSTM의 활성화함수는 softsign을 사용하였고 최적화 함수는 AdamOptimizer, 손실 함수는 평균제곱오차(Root Mean Square Error)를 사용했다. 학습셋과 테스트셋은 학습셋 0.7, 테스트셋 0.3으로 나누어 학습했다.

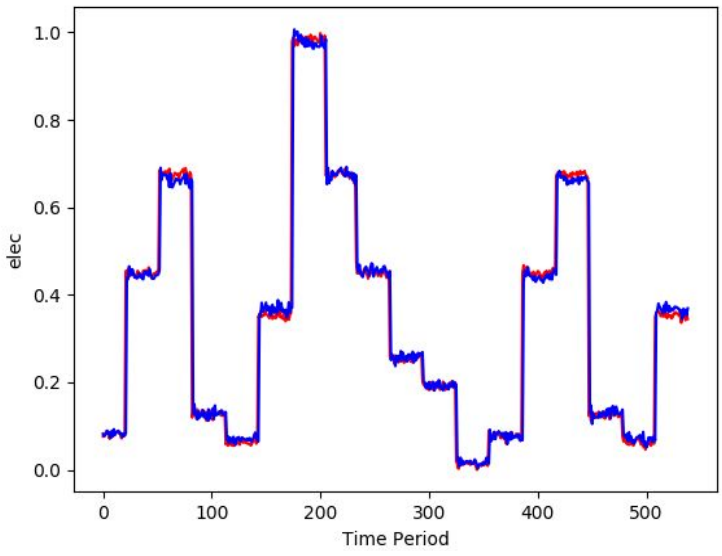


학습을 한 결과 육안상이나 정확도 측면에서 낮게 나와서 seq_length를 한달 기준인 30으로 낮추고 다른 조건을 동일하게 모델을 설정하여 다시 진행했다.

<하이퍼 파라미터>

Input_data_column_cnt : 9	입력데이터의 컬럼 개수
output_data_columns_cnt : 1	결과데이터의 컬럼 개수
seq_length : 30	1개 시퀀스 길이
rnn_cell_hidden_dim : 20	각 셀의(hidden)출력 크기
forget_bias : 1.0	망각편향(기본값:1.0)
num_stacked_layers : 1	stacked LSTM layers 개수
keep_prob : 1	dropout 할 때 keep 할 비율
epoch_num : 1000	에폭 횟수
learning_rate : 0.01	학습률

전과 같이 LSTM의 활성화함수는 softsign을 사용했고 최적화 함수는 AdamOptimizer, 손실 함수는 평균제곱오차(Root Mean Square Error)를 사용했다. 마찬가지로 학습셋과 테스트셋도 동일하게 학습셋 0.7, 테스트셋 0.3으로 나누어 학습했다.



seq_length를 365일, 즉 1년을 했을 때보다 한달 기준인 30일로 했을 때 더 좋은 결과가 나타났다. seq_length를 더 많이 줄여도 정확도에서 크게 변동되지 않아 한 달 기준인 30일로 모델을 결정했다.

6. 프로젝트 결론 및 한계점

6.1 결론

우리는 빅데이터 솔루션 시스템을 설계하여 발생한 로그 데이터를 수집하고 적재, 탐색 및 처리, 그리고 분석까지 일련의 모든 과정을 구현했다. 플럼을 통해서 발생된 로그데이터 읽어 수집하고 수집된 데이터들을 바로 하둡에 적재하거나 가공하여 적재했다. 적재된 대용량 데이터를 신속하고 효과적으로 처리 및 탐색하기 위해 하이브, 임팔라, 스파크를 이용했고 휴라는 웹 UI를 이용하여 접근성과 편의성을 좋게했다. 이렇게 적재된 데이터를 이용하여 R과 파이썬으로 여러가지 분석과 수요예측까지 했다. 또한 이 모든 과정을 쉽게 설치하고 관리해주는빅데이터 자동화 관리도구인 클라우데라를 이용해 사용자가 더욱 편리하게 사용할 수 있게 설계했다.

위와 같이 설계된 빅데이터 솔루션이 처리한 데이터의 양은 다음과 같다.

- 15분단위 전력 생성 데이터 = 100가구 * 15분주기 * 5년 = 17,280,000건 데이터 처리

- 1초단위 전력 생성 데이터 = 100가구 * 1초주기 * 1주일 = 60,480,000건 데이터 처리

성격이 다른 두가지 데이터를 발생시켜 솔루션에 처리하면서 약 7,700만건의 데이터가 솔루션을 통해 수집-적재, 처리-탐색, 분석-응용 과정을 수행하며 파일럿 프로젝트의 역할을 충분히 진행했다고 생각한다.

하드웨어적 문제로 인해 1초 주기 실시간 데이터는 7일치 만 생성해서 테스트를 진행했다. 이 데이터는 100가구를 기준으로 했을때, 1주일에 약 6천만건, 1년일때는 약 10억 건이 넘는 데이터가 발생한다. 이는 굉장히 한정적인 100가구로 샘플로 했음에도 많은 양의 데이터가 수집되는 것을 확인했다. 이를 실제로 적용해보면, '2017년 인구주택총조사' 의 우리나라 가구수는 2,016만8,000가구로 한전은 1초단위 데이터 수집은 데이터의 양이 무한대이기 때문에 실제 스마트미터를 정보 수집을 15분 주기로 설계한 이유이기도 한 것 같다 라는 결론을 내부적으로 내리기도 했다.

이번 프로젝트는 대용량 로그정보를 처리할 수 있는 솔루션을 구현해보자 라는 목표로 시작했고, 솔루션의 적용될 도메인으로 한국전력공사에서 진행하는 스마트그리드의 일부인 가정/주택용 스마트미터의 시나리오를 적용하여 프로젝트를 진행했다. 이렇게 파일럿 프로젝트를 진행을 한 후 느낀점은 빅데이터의 분석만을 중점적으로 다루기보단 빅데이터 모든 처리과정을 위해 솔루션을 설계하고 구현해봄으로써 데이터의 흐름을 구체적으로 알게된 것이 본 프로젝트의 가장 큰 소득이었다. 지금까지의 분석은 잘 수집되고 분석하기 좋게 정제된 데이터만을 사용했다면, 우리는 로그정보의 수집부터 가공-정제, 분석-응용 단계의 역할군을 나눠서 원천데이터를 데이터 마트를 통해 최종 분석단계에서 사용할 수 있도록 처리함으로 빅데이터의 시초인 가장 작은 데이터부터 다루게 된 셈이다. 따라서 본 솔루션은 위에 대입한 스마트미터의 데이터만 처리하는 것이 아닌 대용량으로 발생할 수 있는 다양한 분야의 데이터를 규모에 맞는 가상환경과 HW/SW를 구축하면 원활히 처리가 가능할 것으로 결론을 내리고 프로젝트를 마무리 했다.

6.2 한계점

프로젝트를 진행하면서 가장 큰 시간비중을 차지했던 것은 소프트웨어/하드웨어의 구성이다. 하둡을 중심으로 많은 서브프로젝트들이 서로 호환이 되도록 설정해주는 부분은 예상치 못한 오류와 프로젝트 일정에 차질을 빚게 했다. 그 중 본 빅데이터 솔루션의 가장 큰 핵심인 실시간으로 생성된 데이터를 적재와 동시에 분석하고 고객에게 실시간으로 제공하는 부분이지만 대규모의 데이터를 비동기방식으로 중계해주는 서브프로젝트인 카프카가 어떠한 이유인지 작동이 되지않아 구현하지 못한점이 가장 아쉬운 부분이었다. 카프카를 대체하기 위해 다른 아키텍처인 RabbitMQ, ActiveMQ를 통해서 구현하려고 했지만, 이 두가지 아키텍처는 효율적이며 구성하기 쉽지만 대용량 데이터를 처리하는 기능에서는 단점이 있었다. 또한 플럼을 통해 바로 하둡에 적재하는 방법도 있지만 만약 하둡에 장애가 발생하면 플럼의 채널에 전송하지 못한 데이터들이 빠르게 쌓이면서 곧바로 플럼의 장애로 이어져 데이터 유실이라는 치명적인 문제가 발생하기 때문에 실행하지 못했다. 이러한 오류의 발생과 복잡한 구성을 보다 쉽게 연결하기 위해 클라우데라 매니저를 사용했음에도 불구하고 카프카는 구현되지 않았다.

또한 하드웨어적인 부분에서의 문제점도 발견되었다. 실제 스마트미터는 가정/주택에 모두 설치가 되어 데이터를 수집하지만 파일럿 프로젝트로 100대의 스마트미터를 설치했다는 가정하에 수행을 할 수 있는 가상환경을 구현했다. 하지만 가상서버 3대를 구현하면서 사용하는 PC의 성능문제로 인해 메모리 부족, 공간 부족등으로 데이터를 처리하는 과정에서 많은 문제점을 보였고, 그로 인해 데이터의 양을 줄일 수 밖에 없는 방법을 택하기도 했다.

로그 시뮬레이터 어플리케이션을 개발에 성공했지만, 개발과정은 쉽지 않았고 제한적인 부분이 많았다. 대부분의 빅데이터 프로젝트가 그렇듯이 아무리 프로젝트 시작단계에서 기획안을 철저하게 짜고 시작해도 다시 되돌아오거나 여러가지 변수들 때문에 다른 방향을 가거나 혹은 전체 시나리오를 다시 수정되는 경우들이 많다. 본 프로젝트도 마찬가지였다. 특히 세대별 전력량을 발생시키는 시뮬레이터를 개발할 때 모든 가구들이 똑같은 전력량이 나올순 없으니 가중치 각각 다르게 주어야만 했는데 상당히 복잡한 부분이라서 시나리오의 수정을 거듭하였다. 가중치 부여하는것을 개발하는 사람 마음대로 하는 건 객관적인 자료가 아니기 때문에 한국전력공사에서 2015년에 발표한 보고서를 토대로 월별 그리고 가구원수별 전력사용량의 평균과 표준편차 그리고 비율 등을 계산해서 본 프로젝트에 가중치를 부여하였다. 아쉬웠던 점은 연도별 가중치도 고려하였으나 객관적인 자료를 찾을 수 없었다. 또 스마트미터의 통신 프로토콜을 실제와 똑같이 구현하고 싶었으나 자료가 없어서 구현할 수 없었다는 점도 아쉬운 부분이다.

7.참고문헌

1. 파이썬으로 데이터 주무르기, 저자 민형기

2. 실무로 배우는 빅데이터 기술, 저자 김강원

3. 주택용 전력수요 계절별 패턴 분석과 시사점, 저자 조성진, 윤태형

4. 빌딩 스마트 그리드를 위한 BACnet-ZigBee 기반의 스마트 미터 설계 및 구현, 저자 김형래

웹사이트

1. 기상자료개방포털

(https://data.kma.go.kr/commn/main.do)

2. 클라우데라매니저
(https://docs.cloudera.com/documentation/enterprise/latest/topics/cloudera_manager.html)

3.위키백과(플럼, 하둡, 주키퍼, 임팔라, LSTM)

(https://en.wikipedia.org/wiki/Main_Page)