Apache Kudu와 Impala를 활용한 Lambda Architecture 설계

황은영*, 이길원*, 신용태**
*숭실대학교 컴퓨터학과
**숭실대학교 컴퓨터학부

doublewhy@soongsil.ac.kr, pwlee@soongsil.ac.kr, shin@ssu.ac.kr

Lambda Architecture Design using Apache Kudu and Impala

Yun-Young Hwang, Pil-Won Lee, Yong-Tae Shin
Soongsil University, Science of Computer

요 약
데이터의 양이 기술의 발전으로 발생하는 크게 증가하였고 다양하고 빡빡한 처리 플랫폼이 등장하고 있다. 이 중 가장 널리 사용되고 있는 플랫폼이 Apache 소프트웨어 재단에서 개발한 Hadoop이며, Hadoop은 IoT 분야에도 사용된다. 그러나 기존에 Hadoop 기반 IoT 셀러 데이터 수집, 분석, 환경은 Hadoop의 코어 프로젝트인 HDFS의 Small File로 인한 데이터의 관리 문제와 Import된 데이터의 Update나 Delete가 불가능하다는 문제가 있다. 본 논문에서는 Apache Kudu와 Impala를 활용해 Lambda Architecture를 설계한다. 체계화한 Architecture는 IoT 셀러 데이터를 Cold-Data와 Hot-Data로 분류해 각 성격에 맞는 스토리지에 저장하고 Batch를 통해 생성된 Batch-View와 아파치 Kudu와 Impala를 통해 생성된 Real-time View를 활용해 기존 Hadoop 기반 IoT 셀러 데이터 수집, 분석, 환경의 문제를 해결하고 사용자가 분석할 데이터에 접근하는 시간을 단축한다.

1. 서론
제안하는 Apache Kudu와 Impala를 활용한 Lambda Architecture는 Hadoop 기반의 IoT 셀러 데이터 수집, 분석, 환경에서 발생하는 Small File 문제를 해결한다. 제안하는 Architecture는 Hadoop에서 불가능한 Import된 데이터의 수정 문제를 Apache Kudu와 Impala를 활용하여 해결한다. 경신주기가 짧으며 크기가 작은 실험 데이터와 검증주기가 긴 저장되어 있는 대용량 데이터를 빠르게 분석하는 환경을 제공한다.
본 논문의 구성은 다음과 같다. 2장에서는 Lambda Architecture의 정의와 특성, Apache Kudu와 Apache Impala에 대해 살펴본다. 3장에서는 제안하는 Apache Kudu와 Impala를 활용한 Lambda Architecture를 설계한다. 4장에서는 결론 및 향후 연구 방향을 제시한다.
2. 관련 연구

2-1. Lambda Architecture

Lambda Architecture는 오래된 데이터를 보관하는 배치(Batch) 데이터의 시점과 실시간 데이터를 가진 실시간 데이터를 JOIN하여 결과값을 얻을 수 있도록 구성한 Architecture이다.[2] [그림 1]은 Lambda Architecture의 구조를 나타낸다.

[그림 1] Lambda Architecture의 구조


2-2. Apache Kudu


Apache Kudu는 데이터 저장소 역할만 하는 플랫폼으로 이를 사용하기 위해 서버가 필요하다.[3] [그림 2]는 Apache Kudu의 서버 구성을 나타낸다.

[그림 2] Apache Kudu의 서버 구성

Apache Kudu 내 데이터는 구조화되어 데이터에 저장되며, 테이블은 Tablet이라는 단위로 세분화되어 Tablet 서버에 저장된다. 스토리지 시스템은 메타데이터를 관리하는 마스터 노드와 사용자 데이터인 Tablet을 저장하는 Tablet 서버로 구성된다. Apache Kudu는 하나 이상의 마스터 노드와 Tablet 서버로 구성된다. Apache Kudu는 단순한 CRUD만 제공하기 때문에 복잡한 기술을 사용하기 위한 복잡의 처리가 필요하다.

2-3. Apache Impala

Apache Impala는 HDFS를 위해 Apache 소프트웨어 재단에서 개발한 분산 명령 절의 처리 엔진이다. Apache Impala는 스토리지에 저장되어 있는 데이터를 SQL을 통해 실시간으로 분석하는 시스템으로 스토리지 엔진이 제공하지 않는 연산을 실행한다.[4] Apache Impala는 MapReduce를 이용하지 않는 분산 절의 엔진을 통해 SQL을 실행하여 낮은 연산시간으로 결과를 제공한다. [그림 3]은 Apache Impala의 구성

[그림 3] Apache Impala의 구성
Impala의 구조를 나타낸다.


3. 제안하는 Lambda Architecture 설계


4. 결론

본 논문에서 제안하는 Apache Kudu와 Impala를 활용한 Lambda Architecture는 Bactch View와 Apache Kudu와 Impala로 생성한 Real-time View를 통해 클라이언트가 결과까지 접근하는 시간을 단축하며, Hadoop 기반 데이터 수집 분석 환경에서 발생하는 Small File로 인한 대용량의 과부하 문제를 해결할 수 있다. 향후 본 논문에서 제안하는 Apache Kudu와 Impala를 활용한 Lambda Architecture의 구축이 필요하며, Cold-Data와 Hot-Data를 운영환경에 맞춰 자동으로 분류하는 알고리즘의 연구가 필요하다.

Acknowledgement

이 논문은 2020년도 정부(과학기술정보통신부)의 재원으로 정보통신기술진흥센터의 지원을 받아 수행된 연구임 (No.IITP-2019-0-00135, ICT 기반 환경 모니터링 센서 신뢰성 검증 및 평가 플랫폼)

참고문헌