Journal of The Korea Society of Computer and Information
Vol. 22 No. 1, pp. 49-56, January 2017

www.ksci.re.kr
https://doi.org/10.9708/jksci.2017.22.01.049

# Research on Big Data Integration Method

Jee-Hyun Kim*, Young-Im Cho**

## Abstract

In this paper we propose the approach for big data integration so as to analyze, visualize and predict the future of the trend of the market, and that is to get the integration data model using the R language which is the future of the statistics and the Hadoop which is a parallel processing for the data. As four approaching methods using R and Hadoop, ff package in R, R and Streaming as Hadoop utility, and Rhipe and RHadoop as R and Hadoop interface packages are used, and the strength and weakness of four methods are described and analyzed, so Rhipe and RHadoop are proposed as a complete set of data integration model. The integration of R, which is popular for processing statistical algorithm and Hadoop contains Distributed File System and resource management platform and can implement the MapReduce programming model gives us a new environment where in R code can be written and deployed in Hadoop without any data movement. This model allows us to predictive analysis with high performance and deep understand over the big data.

▸ Keyword : R, Big Data, Hadoop, ff, Streaming, Rhipe, RHadoop

## I. Introduction

최근 정보화 사회 패러다임이 하드웨어에서 소프트웨어에 이어 데이터로 확산됨에 따라 빅 데이터가 그 중심에 핵심으로 자리하고 있다. 이러한 현상은 모바일 스마트 기기의 일상화, SNS의 활성화, 클라우드 컴퓨팅, M2M(Machine to Machine)네트워크의 확산으로 데이터 폭발이 더욱 가속화되어 빅 데이터 기반이 확대되고 있다[1].

빅 데이터의 효율적인 저장과 신속한 분석에 필요한 고성능 컴퓨팅 관련 연구가 활발히 진행되고 있는 가운데 빅 데이터의 저장 및 처리에 대한 대표적인 기술로 구글에서 개발한 GFS(Google File System)와 MapReduce가 있으며, 이를 기반으로 아파치 재단에 의해 개발된 하둡이 있다.

하둡은 규모가 큰 데이터 셋의 분석과 처리에 사용되는 처리 도구중 하나이다. 주요 구성요소로는 데이터의 신뢰성있는 저장매체로 사용되는 하둡 분산파일시스템 HDFS(Hadoop Distributed File System)와 데이터 처리에 사용되는 MapReduce가 있다. 하둡의 MapReduce와 HDFS는 구글 파일시스템과 MapReduce에 있는 구글 기술에 의해 풍부하게 되었다. 하둡은 상용 하드웨어에 내장된 컴퓨팅 클러스터에 매우 큰 데이터 세트의 분산 처리와 분산 저장을 위해 자바로 작성된 분산 오픈 소스 소프트웨어에 의해 구현된다. 실제로 하둡 프레임워크는 자바로 작성되었으나 하둡 프로그램은 Java가 아닌 하둡 Streaming 인터페이스를 사용해서 python이나 C++과 같은 타 언어로 개발될 수 있다[2].

빅 데이터 분석을 위해 구글, 페이스북은 분석엔진으로 R

• First Author: Jee-Hyun Kim, Corresponding Author: Young-Im Cho
*Jee-Hyun Kim (jhkim@seoil.ac.kr), Dept. of Computer Software, Seoil University
**Young-Im Cho (yicho@gachon.ac.kr), Dept. of Computer Engineering, Gachon University
• Received: 2016. 12. 08, Revised: 2017. 01. 04, Accepted: 2017. 01. 26.
• The present research has been conducted by the Research Grant of Seoil University in 2016.

을 사용하고 있으며, 오라클, IBM의 Netezza, SAP, Teradata등에서도 in-memory 혹은 in-database분석 엔진으로 R을 채택하고 있다. 이처럼 R은 빅 데이터 분석의 공통 분석 플랫폼으로 여겨지고 있는데, 이는 R이 최신의 다양한 라이브러리를 제공할 뿐 아니라 Java, C, Python등과의 연동이 용이하기 때문이다[3].

R은 통계 데이터 처리, 그래픽뿐만 아니라 오픈 소스 컴퓨팅의 무료 소프트웨어 환경을 위한 프로그래밍 언어이다.

최근 다양성과 특수성을 가진 독립된 소스와 자율성을 침해하지 않으면서 이종의 데이터로부터 빅 데이터를 통합하려는 경향은 세계적으로 현재 일어나고 있는 통합 프로세스 상황에서 매우 의미 있는 일이다.

데이터 통합의 일반적인 접근은 단지 데이터 레벨에서의 통합에 관해서 개략적인 요약으로 기술되어 있는데, 이러한 경우 통합 프로세스에서 데이터 무결성을 보장할 수 있는 아파치 하둡과 프로그래밍언어 R을 사용하는 것이 일반적이다. 그러나 R과 하둡은 빅 데이터를 처리한다는 점에서는 유사하고 각자의 장점이 있으나 데이터 통합 모델이 제시되지 않아 빅 데이터 통합 시 어떤 강점과 단점이 있는지 파악하지 못함으로써 빅 데이터 분석에 문제점이 발생하고 있다.

본 논문에서는 R과 하둡을 통합하기 위한 네 가지 접근 방법을 제시하고 비교하고자 한다.

본 논문의 구성은 다음과 같다. 2장에서 빅 데이터의 개념 및 R과 하둡의 구조 및 기능에 대하여 기술하고, 3장에서 R과 하둡을 이용한 네 가지 데이터 통합방법을 제시하고, 4장에서 제시한 네 가지 방법을 비교하여 적합한 데이터 통합 모델을 제안하며 5장에서 결론으로 마무리한다.

## II. Related Study

### 1. Issues on Big Data

빅 데이터는 각 기관의 하드웨어와 기반 구조에 따라 전통적인 관계 형 데이터베이스와 소프트웨어 기술을 사용하여 구조화되거나 비 구조화된 처리가 어려운 막대한 양의 데이터를 기술하는데 사용되는 용어이다.

빅 데이터는 IBM에 따르면 Fig. 1과 같이 세 가지 주요 속성을 가진다. 먼저 다양성(Variety)은 구조화나 반 구조화, 비 구조화된 여러 가지 형태의 데이터(텍스트, 오디오, 비디오, 클릭 스트림즈, 로그 파일등)를 뜻한다. 둘째 규모(Volume)는 수백의 테라바이트나 페타바이트의 정보를 말한다. 셋째는 데이터의 비즈니스 가치를 극대화하기 위하여 실시간으로 분석되어져야 할 속도(Velocity)가 있다[4][13][14].

또한, 오라클에서는 빅 데이터의 속성을 IBM의 3V에 가치(Value)를 더하여 4V로 정의하기도 하며, 이외에도 다양한 정의가 있다[5].

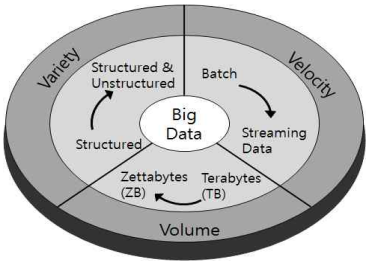


Fig.1. Big data attributes

빅 데이터는 플랫폼 기술에 의해 복합적으로 데이터를 구성한다. 빅 데이터 기술은 복잡성과 가치창출이라는 두 축의 상관관계로부터, 단순분석과 원인분석 차원의 진단(diagnostic analytics)을 통해 통찰력을 갖는 단계인 기본단계와 이로부터 예상분석과 처방분석을 통해 미래에 대한 선견지명을 갖는 연구단계로 크게 두 가지로 구분할 수 있다. 빅 데이터는 기본단계에서부터 연구단계로 점차로 이동하는 것으로 각 단계를 무시하여 혹은 건너뛰어서 미래를 예측할 수는 없다. 또한 빅 데이터 플랫폼은 처리 인프라에 따라 데이터 수집, 데이터 통합, 데이터 저장, 데이터 분석, 데이터 시각화의 5단계로 구성된다[1].

이와 같이 빅 데이터는 다양한 분야에서 다양한 견해로 바라보고 있지만 빅 데이터 관련 기술은 미래의 핵심 기술 중 하나로 기업의 미래 경쟁력을 좌우하는 기반이 될 것이다.

### 2. Open source language R

R은 데이터 모델링, 조작, 통계, 예측, 시계열 분석 및 시각화에 사용되는 오픈 소스 언어이다. R 언어는 작업을 위해 보유한 컴퓨터의 RAM을 사용하며 RAM이 크면 클수록 더 큰 규모의 데이터를 보유할 수 있다.

현재 요구에 따라 사용되기 위해 다양한 학자들에 의해 개발된 5,000개 이상의 통계, 예측분석, 데이터 가시화 패키지가 있다.

이렇게 R은 가장 널리 사용되는 데이터 분석 소프트웨어이며 강력한 통계 프로그래밍 언어로써 유일한 데이터 시각화를 이끄는 오픈소스 언어라는 강점이 있으나, 타 소프트웨어보다 배우기 어렵고 모든 데이터를 메인메모리에 넣으며 많은 수의 패키지 때문에 더 나은 패키지를 찾고 선택하는 것이 힘들며, R 패키지의 수정이 반영되지 않는 단점도 가지고 있다[6].

초기 R은 메모리 한계 문제로 빅 데이터 분석 언어로 사용되지 못했으나 점차 R이 빅 데이터 처리를 위해 ff, ffbase, RODBC, rmr2 and Rhdf와 같은 라이브러리를 가지게 되었다. rmr2와 Rhdf는 함께 빅 데이터를 효과적으로 처리하기 위해 하둡의 기능을 사용한다[4].

### 3. Introduction to Hadoop

하둡은 빅 데이터 분석에 대한 분산 응용을 수행할 자바로 작성된 오픈 소스 아파치 소프트웨어이다[3]. 구성요소로는 하둡 분산파일시스템(HDFS, Hadoop Distributed File System)과 병행처리 배치(batch) 프레임웍이 포함되어 있다. 하둡은 클러스터 내 노드들 간의 신뢰성있는 데이터 이동을 제공하고 있으며, 핵심 기능은 공통 처리 알고리즘인 MapReduce에 있다.

MapReduce는 데이터가 매핑 기능에 기초해서 나눠지고 병행으로 처리하기 위해 클러스터 내의 각 노드들에 적용되는 분할과 정복(Divide & Conquer) 접근 방식을 사용한다. 하둡과 MapReduce는 세 가지 다른 데이터노드(Slave nodes)에 복제된 기본 데이터에 의해 무결점의 고급 레벨을 제공한다[4].

하둡은 Fig. 2와 같이 주 컴퓨터(master)와 종속 컴퓨터(slaves)들을 하나의 클러스터로 묶어 주 컴퓨터를 네임노드, 종속컴퓨터들을 데이터노드라고 부른다. 기능적으로는 크게 HDFS와 MapReduce로 구성되어 있는데, HDFS는 대용량 데이터를 분산저장하고 처리하는 기능을 수행하며, MapReduce는 HDFS에 분산 저장된 데이터를 이용해 병렬 연산을 지원하는 시스템이다. MapReduce 시스템에서는 하둡 클러스터의 주 컴퓨터를 job tracker라 하고 종속 컴퓨터들을 task tracker 라 부른다[7].

아파치 소프트웨어 재단에 따르면 HDFS의 주목적은 네임노드 오류, 데이터노드 오류, 망 분할을 포함하는 오류에서조차 데이터를 신뢰성 있게 저장하는 것이다. 네임노드는 HDFS 클러스터를 위한 오류의 단일 점이고 데이터노드는 하둡 관리 시스템에 데이터를 저장한다.

야후는 4,300 노드이상의 하둡 클러스터를 설정했고 페이스북은 하둡 클러스터에 100PB이상의 데이터를 보유하고 있다[4].

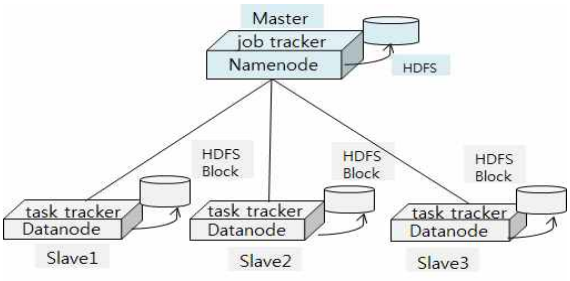


Fig. 2. Architecture of Hadoop Cluster

그러나 R과 하둡은 빅 데이터를 처리하기 위해 주로 사용하기 때문에 빅 데이터 통합처리를 위한 통합모델의 제시가 필요하다.

## III. Data Integration Method

본 절에서는 R의 ff, ffbase 패키지, R과 하둡 Streaming, Rhipe, RHadoop등의 데이터 통합 방법을 제시하고 각 방법들의 비교 분석을 통해 데이터 통합모델을 제안한다.

### 1. Data Integration Model using R and Hadoop

R과 하둡의 통합된 데이터 분석 도구의 일반적 구조는 Fig. 3과 같은 계층구조로 나타낼 수 있다.

첫 번째 계층은 하드웨어 계층으로써 상용 컴퓨터들의 클러스터로 구성되어 있다. 두 번째 계층은 미들웨어 계층으로써 하둡이 여기에 속한다. 이 계층에서는 HDFS와 MapReduce 작업을 이용하여 파일의 분포를 관리한다. 세 번째 계층은 데이터 분석에 대한 인터페이스를 제공하는 계층 이다. 이 계층에 Pig_Latin 이라 불리는 언어를 사용하여 MapReduce프로그램을 작성하는 고급 플랫폼인 Pig와 같은 도구를 가질 수 있다. 또한 하둡의 최상위에 내장되어 있고 아파치에 의해 개발된 데이터웨어하우스 기본 구조인 Hive를 가질 수 있다. Hive는 질의 수행 기능을 제공하고 HiveQL이라 불리는 SQL과 닮은 언어를 사용하여 데이터를 분석하고 MapReduce 작업 구현을 지원하는 기능을 제공한다.

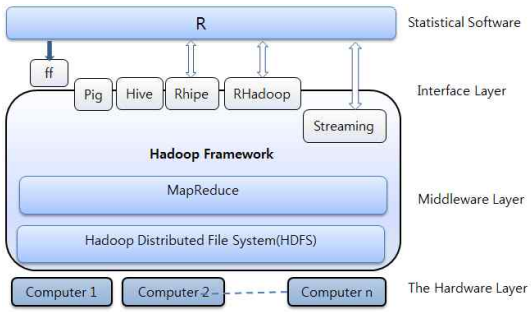


Fig. 3. R, Hadoop and data analysis tools

이 두 가지 도구 외에도 R과 같은 통계 소프트웨어와의 인터페이스를 이 계층에서 구현할 수 있다. 하둡과 R 사이에 인터페이스를 구축하는 Rhipe 이나 Rhadoop 라이브러리를 사용할 수 있는데, 즉 사용자에게 하둡 파일 시스템으로부터 데이터 접근을 허용하고 Map과 Reduce 작업의 구현을 위해 그들 자신의 스크립트 작성을 허용하도록 한다. 또한 R에 통합된 기술인 ff 패키지와 하둡에 통합된 기술인 Streaming을 사용할 수 있다.

본 논문은 R과 하둡 사이 인터페이스로서 사용되는 주요 소프트웨어나 패키지를 분석하여 바람직한 데이터 통합모델을 제안하려는데 있다.

## 2. ff, ffbase package in R

기존의 R은 RAM에 모든 것을 저장한다. R은 하드웨어 구성에 따라 2~4GB까지 메모리를 수용할 수 있다.

ff 패키지는 Daniel Adler외 4인에 의해 개발되었고, Jens Oehlschlagel에 의해 보완되었는데 RAM 메모리 대신 하드디스크, CD, DVD를 사용하여 기본 바이너리 플랫 파일(ff, flat file)을 저장한다. 또한 매우 규모가 큰 데이터 파일을 동시에 작업할 수 있는데, 데이터 파일을 청크(chunk)로 읽고, 외장 드라이브에 그 청크를 저장하는 기능을 한다. 즉, 메모리에 전체 데이터 셋을 로드하는 대신에 한번에 요구하는 데이터의 일부인 청크를 로드하여 메모리 문제를 해결하는 전략을 사용한다[8].

같은 방법으로 청크에 csv나 기타 플랫 파일의 작성이 가능한데, Fig. 4와 같이 HDD나 외부 매체로부터 청크 단위로 읽고 csv나 기타 지원형식으로 그것을 작성한다.

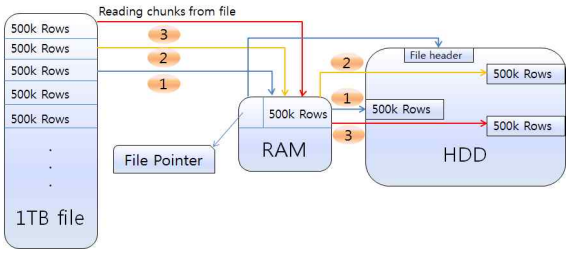


Fig. 4. Functioning of ff Package

ff는 Fig. 5의 예와 같이 조인, 집합, 슬라이싱과 같은 모든 종류의 기능을 구현하는 ffbase 패키지와 함께 편의를 제공한다.

```
File_chunks <- read.csv.ffdf(file="big_data.csv",
 header=TRUE, sep=",", VERBOSE=TRUE,
 next.rows=500000, colClasses=NA)

write.csv.ffdf(File_chunks, "new_file.csv")

Merged_data=merge(ffobject1, ffobject2,
 by.x=c("Col1", "Col2"), by.y=c("Col1", "Col2"), trace=T)

AggregatedData =ffdfdply(ffobject,
split=as.character(ffobject$Col1), FUN=function(x)
summaryBy(Col3 + Col4 + Col5 ~ Col1, data=x, FUN=sum))
```

Fig. 5. Read/Write CSV file using ffdf, join & aggregation

ff는 빅 데이터로 작업하며 RAM에 대한 의존성을 줄이는 장점이 있으나 다음의 몇 가지 한계점이 있다. 첫째 가끔 거대한 데이터 셋의 복잡한 작업을 수행할 때 속도가 문제가 된다. 둘째 ff를 사용한 개발이 쉽지 않다는 것이며, 셋째 디스크에 저장하는 플랫 파일에 주의가 필요한데 HDD나 외장 매체에 공간이 별로 남아 있지 않을 수 있다는 것이다.

## 3. R and Hadoop Streaming

하둡 Streaming은 하둡 배포시 제공되는 유틸리티로써 사용자가 mapper나 reducer로써 실행 파일이나 스크립트를 이용해 Map/Reduce 작업을 생성하고 수행하도록 한다.

Streaming은 사용자가 Map/Reduce 작업을 어떤 스크립트 또는 표준입력에서 데이터를 읽고 mapper나 reducer로써 표준 출력으로 결과를 작성하도록 허용하는 하둡 배포에 통합된 기술이다[9]. 이것은 R이 표준입력으로부터 읽고 쓸 수 있으므로 단계를 삭제하고 map에서 Streaming과 R 스크립트를 함께 사용할 수 있게 한다. 이러한 접근에서 사용자는 R 스크립트의 mapper와 reducer를 지정하는 인수와 함께 Streaming 작업을 수행하기 위해 하둡 명령 라인을 사용할 것이므로 R의 클라이언트들과의 통합은 없다.

R 스크립트로 구현된 map-reduce 작업의 명령라인이 Fig. 6에 나타나있다.

```
$ $ {HADOOP_HOME}/bin/JADOOP jar $ {HADOOP_HOME}/contrib/streaming/*.jar \
-inputformat.org.apache.hadoop.mapred.TextInputFormat \
-input input_data.txt \
-output output \
-mapper /home/tst/srd/map.R \
-reducer /home/tst/src/reduce.R \
-file /home/tst/src/map.R \
-file /home/tst/src/reduce.R
```

Fig. 6. An example of a Map-reduce task with R and Hadoop integrated by Streaming framework

Streaming을 사용한 R과 하둡의 통합은 쉬운 작업이다. 왜냐하면, 사용자는 단지 명령어 라인 인자로써 mapper와 reducer를 지정하는 Streaming작업을 수행하기 위해 하둡 명령어 라인을 수행할 필요가 있기 때문이다. 이러한 접근은 단순한 작업이지만 R이 하둡 클러스터의 모든 데이터노드에 설치되어져 있어야 한다.

또한 이 접근에 대한 라이선스 방식은 하둡에 대한 Apache2.0 라이센스와 R에 대한 GPL-2와 GPL-3의 결합이 필요하다.

## 4. RHIPE

Rhipe은 "R and Hadoop Integrated Programming Environment"를 뜻하며 R 과 하둡 간의 긴밀한 통합을 제공하는 오픈 소스 프로젝트이다. 즉, 사용자가 R에서 직접 빅 데이터의 분석을 수행하고 R 사용자에게 자바 개발자처럼 하둡의 기능을 제공하도록 허용한다.

Rhipe의 설치는 사용자가 각 데이터노드에 R, 구글 프로토콜 버퍼, R과 하둡의 경로 설정, 그리고 Rhipe을 설치해야하는데 그 절차가 매우 까다롭고 복잡한 단점이 있다. R이 각 노드에서 공유 라이브러리로 내장되어야 하고 구글 프로토콜 버퍼가 각 노드에 설치되고 내장되어야 하며, 마지막으로 Rhipe 자체를 설치해야 한다. 프로토콜 버퍼는 효율성을 증가시키고 타 언어와의 상호 운영성을 제공하는 데이터 연속성을

위해 필요하다[9].

Rhipe은 R에서 MapReduce 작업의 수행을 허용하는 R 라이브러리이다. 사용자는 특정 기본 R map과 reduce 기능을 작성해야하고 Rhipe이 나머지를 관리한다. 즉 Rhipe은 map과 reduce 기능을 전송하고 map과 reduce 작업으로부터 그들을 호출한다. map과 reduce입력은 map과 reduce기능을 호출하기 위해 R을 사용하는 Rhipe C 라이브러리로, 프로토콜 버퍼 인코딩 방식을 이용해서 전송된다. 병행 R 패키지가 아닌 Rhipe 사용의 잇점은 하둡과의 통합에 있는데, 하둡은 프로세서 사용을 최적화하려 시도하고 무중단 시스템을 제공하는 컴퓨터 클러스터를 하둡 분산파일시스템(HDFS)을 사용하여 데이터 분산 방식을 제공하기 때문이다.

Rhipe을 사용하는 R 스크립트의 일반적 구조가 Fig. 7에 나타나있고 스크립트를 작성하는 것은 매우 간단하다.

Rhipe은 사용자가 데이터 처리 알고리즘에 집중하도록 하고 컴퓨터 클러스터에 대한 데이터 분산과 계산은 Rhipe과 라이브러리, 하둡에 의해 처리되도록 한다.

```
library (Rhipe)
rhinit (TRUE, TRUE);
map<-expression ({lapply (map.values, function (mapper) . . . })}
reduce <-expression {
pre = { . . . },
reduce = { . . . },
post = { . . . },
x <- rhmr (
map=map, reduce=reduce,
                ifolder=inputPath,
                ofolder=outputPath,
                inout=c ('text', 'text'),
                jobname=' a job name'))
rhex (z)
```

Fig. 7. The Structure of an R script using Rhipe

Rhipe은 사용자가 데이터 처리 알고리즘에 집중하도록 하고 컴퓨터 클러스터에 대한 데이터 분산과 계산은 Rhipe과 라이브러리, 하둡에 의해 처리되도록 한다.

이러한 특성으로 인해 Rhipe은 사용자 친화성, 규모가 큰 데이터 셋의 처리, 통계 알고리즘의 적용, 큰 규모의 데이터 시각화 등에서 ff, Streaming, RHadoop 과 비교해 뛰어난 이점을 가지고 있는 것으로 평가하고 있다[18].

### 5. RHadoop

RHadoop은 하둡 플랫폼 위에서 R 프로그램을 수행시키는 것으로 미국의 Revolution Analytics에 의해 개발된 오픈소스 프로젝트이다 [10][16].

RHadoop의 설치는 복잡한 작업은 아니지만 RHadoop은 다른 R 패키지에 종속성을 갖는다. RHadoop 작업은 하둡 클러스터의 각 데이터노드에 종속된 R 과 RHadoop 패키지의 설치를 의미한다.

RHadoop은 rmr, rhdfs, rhbase등의 함수를 제공하는 3R 패키지의 집합이다. rmr은 R에서 하둡 MapReduce기능을 위한 함수를, rhdfs는 R에서 HDFS 파일 관리를 위한 함수를, rhbase는 R에서 분산 DB인 Hbase 관리를 지원하는 함수를 제공한다[4].

RHadoop은 사용자 정의 map과 reduce R 기능을 호출하는 Streaming으로부터 호출된 R 스크립트의 묶음을 갖는다. RHadoop은 사용자가 map과 reduce 작동을 정의하도록 허용하는 Rhipe과 비슷하게 수행한다. RHadoop을 사용하는 스크립트가 Fig. 8과 같다.

```
library (rmr)
  map<-function (k, v) { ...}
  reduce<-funtion (k, vv) {...}
  mapreduce (
input = "data.txt",
                output="output",
                textinputformat = rawTextinputformat,
map = map, reduce=reduce)
```

Fig. 8. The Structure of an R script using RHadoop

MapReduce 프레임웍은 하둡의 신경 시스템이다. MapReduce는 분할 정복(Divide and Conquer) 접근방식을 사용하여 병렬로 수행한다.

rmr이 map과 reduce 기능에 적용할 수 있는 클라이언트 측 R환경을 만든다. 이 접근에 필요한 라이선스 방식은 하둡과 RHadoop에 대한 Apache 2.0 라이선스와 R에 대한 GPL-2 와 GPL-3의 결합을 의미한다.

## Ⅳ. Analysis of Data Integration Method

지금까지의 분석결과를 바탕으로, 많은 양의 데이터 처리 및 분석에 R과 하둡을 이용하는 네 가지 방법에 대하여 설치부터 적용 기술, 특징들에 대하여 비교하여 제시한다.

네 가지 방법중 ff는 R의 단점을 보완한 R 패키지이며 R과 Streaming은 R과 하둡의 Streaming 유틸리티를 사용한 것이고, Rhipe과 RHadoop은 R과 하둡 사이의 인터페이스 패키지이다.

즉, R만 사용한 방식인 ff, R과 하둡 유틸리티를 사용한 방식인 R Streaming, R과 하둡의 통합 환경을 이용한 인터페이스 패키지 방식의 Rhipe, RHadoop 등의 강점과 약점을 비교분석하여 적합한 데이터 통합모델을 제안하고자 한다.

Table 1의 비교표에 의하면 R의 설치는 ff 패키지와 R과 Streaming에서는 쉬운 작업이나 Rhipe와 RHadoop에서는 하둡 클러스터의 설정을 위한 노력이 요구되어 복잡한 단점이 있다.

ff와 Streaming은 사용자 친화적인 장점이 있으나, Rhipe, RHadoop은 사용자가 접근하는데 절차상의 지식이 필요하여 불편한 점도 있다.

규모가 큰 데이터 셋의 처리는 ff는 약간의 제약이 있으며

Streaming, Rhipe, RHadoop 은 모두 가능하다.

통계 알고리즘의 적용은 ff, Rhipe, RHadoop 모두 가능하나 Streaming은 하둡의 유틸리티를 이용하므로 약간의 제약이 있다.

규모가 큰 데이터의 시각화는 ff, Streaming에서는 제약이 있고 Rhipe, RHadoop에서는 가능하다.

사용자 친화성에 있어서는 4가지 방법 각각의 설치와 이해 및 접근의 용이성을 중심으로 평가하였고 규모가 큰 데이터 셋의 처리는 하둡의 빅 데이터 처리 성능을 고려하여 하둡의 사용 여부가 평가에 참작되었다. 통계 알고리즘의 적용은 R의 검증된 통계 알고리즘의 적용이 용이한지에 근거를 두었으며 규모가 큰 데이터 시각화는 하둡의 큰 규모 데이터 처리의 장점과 R의 시각화 기능을 적극적으로 사용할 수 있는 패키지를 염두에 두고 분석한 결과이다.

기술적인 요구가 ff에서는 빅 데이터를 읽고 쓰는데 필요한 청크의 설계가 필요하며, Streaming, RHadoop은 Streaming 기술이 요구되고 Rhipe은 구글 프로토콜 버퍼를 가지는 자체 기능이 필요하다.

R의 클라이언트측 통합은 ff, Streaming에서는 없고, Rhipe, RHadoop 에서는 높다.

MapReduce 작업 솔루션이 ff는 R 패키지이므로 없고 Streaming에서는 텍스트입력 데이터 파일로 제한하여 가능하며, Rhipe, RHadoop에서는 복잡한 작업도 가능하다. Streaming이 map과 reduce 기능을 인자로 보내는 명령 라인의 접근을 사용하는 동안 Rhipe과 RHadoop은 사용자에게 R 내에서 그들 자신의 map과 reduce 기능을 정의하고 호출하게 한다.

마지막 제약조건으로 R의 ff는 플랫 파일의 용량에 주의하여 HDD나 외장 매체의 공간이 부족하지 않도록 주의해야 하며, Streaming, Rhipe, RHadoop은 R이 하둡 클러스터의 모든 데이터노드에 설치되어 있어야 한다는 제약이 있다.

R 패키지인 ff는 R의 장점인 설치가 간단하고 사용자 편의성은 좋으나 데이터 사이즈에 따른 외장 매체의 공간에 제약이 있으며, R과 하둡 Streaming의 사용은 R의 장점인 설치의 용이성과 사용자 편의성을 가지고 하둡의 장점인 규모가 큰 빅 데이터 처리는 용이하나 MapReduce 작업이 텍스트 입력 데이터 파일로 제한된다는 단점이 있다.

Rhipe과 RHadoop은 R과 하둡의 통합된 프로그래밍 환경을 제공하는 오픈소스 프로젝트로 RHadoop은 각 데이터노드에 R과 RHadoop패키지의 설치로 비교적 설치가 수월하나 Rhipe은  각 데이터노드에 R과 구글 프로토콜 버퍼, Rhipe의 설치가 필요하여 설치 작업이 쉽지 않은 단점이 있다.

그럼에도 Rhipe과 RHadoop의 이점은 규모가 큰 데이터 셋의 처리, 통계 알고리즘의 적용, 데이터 시각화가 가능하며, 중요한 분석을 수행하고 데이터를 효과적으로 처리할 수 있는 완전체로써 복잡하고 어려운 빅 데이터의 통합 모델로 제안한다.

Table 1. Comparison of Data Integration Method

| | ff | R & Streaming | Rhipe | Rhadoop |
|---|---|---|---|---|
| R installation | easy | easy | need to set up cluster (high) | need to set up cluster (medium) |
| User friendly | high | high | medium | medium |
| Handle Large data set | medium | possible | possible | possible |
| Apply Statistical Algorithm | possible | medium | possible | medium |
| Large data visualization | medium | medium | possible | medium |
| Technology | ff package | Streaming | Google Protocol Buffers to be built | Streaming |
| Memory Usage | RAM,HDD,CD, DVD | | | additional memory for reducer |
| client-side integration with R | none | none | high | high |
| licensing scheme | combination of GPL-2, 3 | combination of GPL2,3 Apache 2.0 | Apache 2.0 | Apache 2.0 |
| MapReduce job solution | none | direct sol.-limited in text file | possible complicate job | possible complicate job |
| Specification | flat file not to be over external media storage | R to be installed on every DataNode of the Hadoop cluster | R to be installed on every DataNode of the Hadoop cluster | R to be installed on every DataNode of the Hadoop cluster |

Fig. 9는 위의 비교에 대한 실험적 검증을 위하여 R과 ff의 메모리 사용량을 조사한 그래프이다[17]. 이 자료를 바탕으로 R의 시간 t에서의 메모리 점유율을 각각 데이터공간, 식별공간, 결과공간으로 구분하여 비교하였다. 비교결과, R은 ff 패키지에 비해 시간 t에서 5.7배 만큼의 메모리 점유율을 나타내고 있으며, ff패키지는 13.2%의 총 메모리대비 점유율을 나타내며, R은 74%의 점유율을 나타내고 있다. 이 결과는 ff 패키지가 R에 비해 메모리 관점에서 훨씬 효율적으로 사용하도록 설계됨을 보여주는 것으로 R의 메모리 사용 한계를 극복하여 규모가 큰 데이터의 처리나 통계 알고리즘의 적용, 시각화가 가능하도록 하였다.

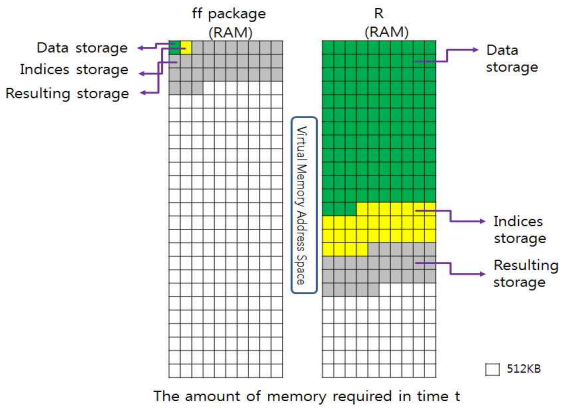


Fig. 9. Comparison of memory usage between ff & R

본 비교분석에서 Fig. 9는 정량적 분석을 하였으나 Table 1의 각 항목에 대하여는 정성적 비교 분석으로 제한되어 있어, 그에 대한 실험적 검증이 필요하며, 향후 정량적 결과를 얻기 위해 관련 프로그램 개발 등 좀 더 면밀한 준비와 분석이 요구된다.

## Ⅴ. Conclusion

IT의 흐름이 소프트웨어에서 데이터로 이동하고 있으며 데이터 소스, 형태의 다양성 및 적시에 데이터를 수집, 처리, 분석하기 위해 빅 데이터에 대한 관심이 고조되고 그에 대한 도구와 기술의 개발은 핵심 이슈가 되고 있다.

본 논문에서는 대규모 데이터 셋의 처리에 R과 하둡을 이용하는 네 가지 방법으로 R에서의 ff, R과 하둡 Streaming, Rhipe, RHadoop을 제시하였고 각 접근들의 이점과 제약조건을 비교 기술하였다. R의 라이브러리인 ff패키지는 빅 데이터 작업과 RAM에 덜 의존한다는 모든 종류의 이점에도 불구하고 여전히 메모리 제약이 있는 중간 계층과 임시파일에 의존한다는 것이다. Streaming과 함께 사용하는 R은 설치에 관해 아무런 문제도 발생하지 않지만, Rhipe과 RHadoop은 클러스터를 설정하기 위해 약간 복잡한 절차를 거쳐야 한다.

ff는 R패키지로 하둡의 Map-Reduce 솔루션은 없으며, Streaming은 간단한 Map-Reduce 작업을 위한 직접적인 솔루션이 있는데, 이 솔루션은 단지 텍스트 입력 데이터 파일로 제한된다. 더 복잡한 작업에 대한 솔루션은 Rhipe 또는 RHadoop 이어야 한다. ff는 저장하는 플랫 파일의 용량에 주의하여 HDD나 외장 매체에 공간이 부족하지 않도록 해야 하며, R과 Streaming, Rhipe, RHadoop은 R이 하둡 클러스터의 모든 데이터노드에 설치되어 있어야 하는 단점이 있다.

제안한 Rhipe과 RHadoop은 규모가 큰 데이터의 처리, 통계 알고리즘의 적용, 데이터 시각화가 가능하며, 빅 데이터의 의미있는 분석을 수행하고 데이터를 효과적으로 처리할 수 있는 통합모델이나 설치 및 설정의 어려움은 해결해야 될 과제이다.

빅 데이터 분석을 위해 데이터를 통합할 수 있는 다른 방법으로 R과 데이터베이스의 연동인 RODBC, RJDBC, Rhive 등과 Revolution Analytics에서 제공하는 R의 상용 버전, R에 대한 Oracle 커넥터등이 있다. 향후 빅 데이터 분석의 용이한 접근을 위한 데이터 수집 및 통합 방법이 개발되어 빅 데이터 분석의 일반화 및 효율성이 증대되기를 기대한다.

## REFERENCES

[1] Young-Im Cho, “Understanding Big Data and Its Main Issue,” Journal of The Korean Association for Regional Information Society, Vol.16, No.3, pp.43-65, September 2013.

[2] Piyush Gupta, Pardeep Kumar, Girdhar Gopal, “Sentiment Analysis on Hadoop with Hadoop Streaming,” International Journal of Computer Applications, Vol.121, No.11, July 2015.

[3] Youngjun Ko, Jinseog Kim, “Analysis of big data using Rhipe,” Journal of the Korean Data & Information Science Society, Vol.24, No.5, pp.975-987, August 2013.

[4] Anju Gahlawat, “Big Data Analysis using R and Hadoop,” IJCEM International Journal of Computational Engineering & Management, Vol.17 No.5, September 2014.

[5] Jean-Pierre Dijcks, “Oracle: Big Data for the Enterprise,” An Oracle White Paper, pp3-4, June 2013.

[6] “R Tools Evaluation,” Telefonica, May 2015. http://madrid.r-es.org/wp-content/uploads/2015/05/R-Evaluation-2015.pdf

[7] Hedlund, B, “Understanding Hadoop clusters and the network,” http://bradhedlund.com/2011/09/10/Understanding-Hadoop-clusters-and-the-network/.

[8] D. Adler, O. Nenadic, W. Zucchini, C. Glaser, “The ff package: Handling Large Data Sets in R with Memory Mapped Pages of Binary Flat Files,” Institute for Statistics and Econometrics, August 2007.

[9] “Hadoop Streaming”, The Apache Software Foundation, 2007. https://svn.apache.org/repos/asf/Hadoop/common/tags/release-0.18.2/docs/streaming.pdf,

[10] “The Revolution Analytics Perspective on Big Data,” http://www.revolutionanalytics.com

[11] “Big Data: the new ‘The Future’,” EPIC 2015. http://www.columbia.edu/~sjm2186/EPIC_R/EPIC_R_BigData.pdf

[12] Manyika J., Chui M., Brown B., Bughin J., Dobbs R., Roxburgh C. and Byers A., “The next frontier for innovation, competition and productivity,” McKinsey & Company, 2011.

[13] Zikopoulos P., Eaton C., Roos de D., Deutsch T. and Lapis G., “Understanding big data: Analytics for enterprise class Hadoop and streaming data,” McGraw-Hill, 2012.

[14] David Corrigan, “Integrating and governing big data,” IBM Software, White paper, January 2013.

[15] SungWoo Jang, “Oracle: Hadoop Based Processing,” Oracle Korea, 2012. http://www.columbia.edu/~sjm2186/EPIC_R/EPIC_R_BigData.pdf

[16] “Implementation of MapReduce in R,” DBguide.net, 2016, http://www.dbguide.net/db.db?cmd=view&boardUid=187501&boardConfigUid =9&categoryUid = 216&boardIdx=162&boardStep=1

[17] Choonghyun You, “Technology Trends in Big Data Analytics and Introduction to R,” NexR, Data Science Team, KRNet2012.

[18] Kevin, “Large Data Analysis Using Rhipe/RHadoop,” ebay, Behavioral Insights and Science Team, Nov. 2013.

## Authors

Jee Hyun Kim received a Doctor of Computer Science from Dankook University, Korea, in 2004, her M.B.A degree in the Information Management from Dankook University in 1994, her B.S degree in Mathematics from Ewha Womans University in 1978. Her major is Software Engineering. She is a professor in the Department of Computer Software in Seoil University. Her research interests are Web Engineering, Big data, Quality Management, Information Retrieval etc.

Young Im Cho received her B.S., M.Sc., and Ph.D from the Department of Computer Science, Korea University, Korea, in 1988, 1990 and 1994, respectively. She is a professor at Gachon University. Her research interest includes AI, Big data, information retrieval, smart city etc.