

UX Analysis for Mobile Devices Using MapReduce on Distributed Data Processing Platform

Sungsook Kim[†] · Seonggyu Kim^{**}

ABSTRACT

As the concept of web characteristics represented by openness and mind sharing grows more and more popular, device log data generated by both users and developers have become increasingly complicated. For such reasons, a log data processing mechanism that automatically produces meaningful data set from large amount of log records have become necessary for mobile device UX(User eXperience) analysis. In this paper, we define the attributes of to-be-analyzed log data that reflect the characteristics of a mobile device and collect real log data from mobile device users. Along with the MapReduce programming paradigm in Hadoop platform, we have performed a mobile device User eXperience analysis in a distributed processing environment using the collected real log data. We have then demonstrated the effectiveness of the proposed analysis mechanism by applying the various combinations of Map and Reduce steps to produce a simple data schema from the large amount of complex log records.

Keywords : Mobile Device Log, User eXperience, MapReduce

MapReduce 분산 데이터처리 플랫폼에 기반한 모바일 디바이스 UX 분석

김 성 숙[†] · 김 성 규^{**}

요 약

웹의 특징인 개방과 공유의 사고방식이 더욱 일반화 되면서 개발자 뿐 만 아니라 사용자가 직접 발생시키는 데이터도 복잡적으로 늘어나고 있는 실정이다. 이러한 상황에서 모바일 디바이스 User eXperience(UX) 분석에서 다른 무엇보다도 디바이스에 기록되는 대용량의 로그 기록에서 필요한 데이터들을 자동으로 요약 정리해 주는 기법이 필요하다. 이에, 본 논문에서는 분석하고자 하는 모바일 디바이스 특성에 맞게 사전에 로그 데이터 속성에 대한 정의를 먼저하고, 직접 이를 반영한 사용자의 로그를 수집하여 저장하였다. 또한, 발생하는 대용량의 로그 기록에 기초한 UX를 분석하고자 다양한 로그 데이터 타입을 설정 및 처리할 수 있는 Hadoop(하둡)에서 제공하는 MapReduce 기법을 활용하여 데이터를 분산 처리하였다. 이를 통해, Map과 Reduce의 다양한 조합으로 대용량의 모바일 디바이스에서 발생하는 로그 데이터 셋에서 복잡한 스키마를 단순화시켜 분산 데이터 처리 환경에 맞게 UX 분석 방안을 제시하였다.

키워드 : 모바일 디바이스 로그, User eXperience, MapReduce

1. 서 론

최근 IT 환경은 모바일 정보 기기의 확산으로 다양한 디지털 데이터가 빠르게 생성되고 있으며, 웹 서비스 확장 및 다양한 기기들 간의 정보 교류로 인해 대용량의 데이터가 기

하급수적으로 축적되고 있다[1]. 모바일 환경에서는 하드웨어적인 기능과 소프트웨어적인 기능들을 고려하고 이에 반응하는 사용자의 상태와 동작 등이 분석에 반영되어야 한다.

모바일 디바이스 User eXperience(이하 UX)에 대한 분석은 먼저 모바일 특성에 맞는 데이터 세트의 속성 정의 단계부터 시작한다. 특히, 주요한 의사 결정에 사용되고 있는 로그데이터는 수많은 하드웨어와 소프트웨어의 유기적인 결합으로 분석 처리에 있어서 기존의 관계형 데이터베이스(이하 RDBMS)의 경직된 데이터 스키마의 표현으로는 한계가 있다[2].

MapReduce는 키/값을 이용한 처리로 분산 저장 환경에

* This study was supported by the R&D Center for Valuable Recycling (Global-Top Environmental Technology Development Program) funded by the Ministry of Environment.(Project No.GT-12-C-01-320-0).

† 준 회 원 : 안양대학교 컴퓨터공학과 박사과정

** 정 회 원 : 안양대학교 컴퓨터공학과 교수

논문접수 : 2013년 7월 16일

수정일 : 1차 2013년 8월 12일

심사완료 : 2013년 8월 12일

* Corresponding Author : Seonggyu Kim (sgkim@anyang.ac.kr)

서 데이터를 병렬 처리할 수 있도록 지원하는 프로그래밍 기법이다. 본 논문에서는 사용자의 UX를 파악하기 위하여 대용량의 모바일 디바이스 로그 분석에 분산 저장 병렬 처리 컴퓨팅 시스템인 하둠을 사용한다. 분산처리 시 하둠이 제공하는 다양한 기능을 이용하여 처리자가 직접 데이터 처리에 대한 다양한 접근을 시도 할 수 있다. 본 연구에서는 MapReduce로 처리할 수 있는 여러 프로그래밍 모델 패턴 중 Numerical Summarization Pattern을 선택하여 중간 정렬을 위한 단계 생략으로 Map의 대용량 입력데이터에 대하여 축소된 Reduce의 출력 데이터를 바로 얻음으로써 대용량의 모바일 데이터 분석 시 처리 향상을 도모할 수 있다[3].

본 논문은 모두 5장으로 구성되어 있으며, 2장에서는 모바일 환경의 변화와 모바일 로그데이터의 특징을 기술하고 디바이스 UX 분석의 의미, 분석에 사용되는 MapReduce의 장점에 대하여 설명한다. 3장에서는 모바일 로그 분석 프로세스의 전체적인 과정과 각 단계별 데이터 처리 분석 과정에 대하여 설명한다. 4장에서는 수집된 로그를 이용하여 MapReduce로 처리한 분석 결과를 보여준다. 마지막으로 5장에서는 결론과 후속 연구에 대해 언급한다.

2. 관련 연구

2.1 모바일 환경의 변화

최근 모바일 환경의 특징은 기존 데스크톱이나 노트북과 같이 여러 기능을 자체적으로 제공함과 동시에 다양한 애플리케이션들을 사용자가 원하는 대로 설치 및 삭제할 수 있는 환경으로 변하고 있다는 점이다. 또한, 여러 통신망을 이용하여 인터넷 서비스를 자유롭게 사용함으로써 기존의 장소에 대한 제약을 어느 정도 벗어났다고 할 수 있다. 따라서 이러한 인터넷 환경에서 사용자들은 원하는 애플리케이션을 직접 제작할 수 있으며, 기존 애플리케이션도 자신에게 맞게 인터페이스를 새롭게 구성할 수 있고, 이로 인하여 개발자와 사용자의 구분이 점점 모호해 지고 있다.

소프트웨어적인 관점에서의 혼재된 모바일 앱의 종류를 보면 네이티브 앱(Native App), 모바일 웹(Mobile Web), 하이브리드 앱(Hybrid App)으로 나눌 수 있다[4]. 네이티브 앱은 특정 모바일 기기에서만 설치 사용할 수 있는 애플리케이션으로 일반적으로 말하는 ‘앱(App)’이다. 모바일 웹은 웹 기술을 모바일 환경에 최적화한 웹서비스 기술이며, 모든 모바일 브라우저에서 같은 형태로 실행되고 보여줄 수 있게 하는 미래 표준화 기술의 대상이기도 하다. 하이브리드 앱은 모바일 웹을 통해 기기를 직접 제어하거나 실행해서 네이티브 앱과 같은 기능을 제공해 주는 앱을 말한다.

이러한 복잡한 모바일 환경의 소프트웨어적인 변화는 하드웨어적인 기술 변화와 맞물려 사용자의 UX 분석을 매우 어렵게 하는 요인이 되고 있다.

2.2 모바일 환경에서의 UX 중요성

UX는 사용자가 시스템, 제품 또는 서비스를 이용하면서

직/간접적으로 알게 되는 총체적인 경험을 의미한다[5]. 모바일 환경은 과거에는 힘들었던 모바일 웹 기술의 발달과 복잡한 네트워크 인프라 확장에 힘입어 구현되는 다양한 소프트웨어 플랫폼들의 연동성으로 모바일 디바이스의 UX 분석을 더욱 복잡하게 하고 있다. 구체적으로 모바일 애플리케이션 플랫폼과 모바일 웹 플랫폼 기술, 이를 지원하기 위한 네트워크 서비스 API 기술 등이 혼재되어 있는 상황이라 할 수 있다.

이러한 상황에서 모바일 디바이스의 하드웨어적인 특성과 소프트웨어적인 특성을 고려한 UX 분석은 사용자의 사용 패턴을 제품에 신속히 반영할 수단이 되고 있다. 또한, 기존의 PC 환경에서와 다르게 플랫폼에 대한 표준이 아직 정착되지 못하고 있는 상태에서 웹 사용이 빠르게 확산되면서 모바일 디바이스에서 발생하는 자료의 복잡성은 더욱 커지고 있다. 요즘은 기기 사용자들의 다양한 애플리케이션 사용으로 이에 대한 실생활의 사용 행태를 조사하는 방법으로는 복잡한 형태의 로그기록에서 수집한 데이터를 다양한 방법으로 분석해야할 필요성이 더욱 커지고 있다.

디바이스 사용 로그란 사용자의 행동 또는 제품의 동작 내용을 시간과 함께 차례로 기록한 것을 말하는데 인간의 행동을 분석하고자 하는 실험 연구에서는 매우 흔히 사용되는 자료이다. 로그는 단순한 트래픽의 양적인 정보 뿐만 아니라 사용자의 모바일 기기 사용 성향에 따른 모든 정보 등을 담고 있기 때문에 실시간으로 기록된 사용자의 UX 분석을 하는데 있어서는 정성적인 방식에 의존했던 기존의 방식보다는 보다 더 정량적인 분석을 요할 수 있는 매우 중요한 자료가 된다.

2.3 하둠에서의 분산 데이터 처리

구글에서 2004년 Google File System[6]과 MapReduce[7] 관련하여 발표한 논문의 기법을 적용하여 개발된 Hadoop(이하 하둠)은 저가 기기를 연결하여 분산 병렬 처리 가능하게 한 Scale out(이하 수평적 확장)에 적합한 컴퓨팅 환경이다. 하둠은 HDFS라는 별도의 파일 시스템을 가지고 있고, 이는 대용량 데이터 처리가 필요한 경우에 최적화되어 있다. 하둠 프레임워크에서 데이터 처리 부분은 MapReduce 프로그래밍 모델이 담당한다[8].

하둠의 분산 파일 시스템인 HDFS와 분산 처리 프레임워크인 MapReduce는 물리적으로 같은 서버에 공존한다. 하나의 마스터와 다중 슬레이브 구조를 갖는데 다중 슬레이브의 경우에도 각 서버마다 HDFS 슬레이브와 MapReduce 슬레이브가 같이 놓임으로써 데이터가 있는 곳으로 코드를 이동하여 분산 처리의 효율을 강화시키고 있다[9]. 즉, 분산 저장소를 이용하고 데이터 대신 코드를 전송하여 처리함으로써, 대용량의 데이터를 전송함으로써 발생하는 여러 병목 현상들을 손쉽게 해결할 수 있다.

이는 다수의 저렴한 서버를 사용하여 수평적 확장한 컴퓨팅 환경에 적합하며, 오류에 대한 대처도 유연성 있게 대처할 수 있다[10,11]. MapReduce의 장점은 먼저 대용량의 데이터 처리를 목적으로 한다는 것과 이를 위하여 분산 환경

에서 프로그램이 처리된다는 점이다. 여기서 각 처리 단계는 독립적이기 때문에 병렬로 실행될 수 있으며 따라서 총 컴퓨팅 시간을 대폭 줄일 수 있다. 또한, 대규모 데이터 병렬 처리를 위해서는 데이터 증가에 따른 확장성을 제공할 수 있어야 하며, 노드 간 데이터 이동에 따른 네트워크 트래픽을 최소화 할 수 있도록 업무 분산이 필요한데 MapReduce는 이와 같은 요구사항을 잘 고려하여 만들어진 프레임워크라 할 수 있다.

따라서 대용량 데이터 처리에 있어서 확장성이 용이한 MapReduce의 분산 병렬성과 RDBMS의 쿼리 개념을 혼합한 다양한 시도가 진행되었다[12,13]. 하지만, 데이터 저장의 아키텍처를 바꾸는 이러한 시도는 데이터의 분산성을 높이는 데는 기여했지만 쿼리에 대한 새로운 연산 처리가 필요하여 성능에 있어서는 기존의 RDBMS 보다 큰 효과를 거두지는 못했다. 또한, 대용량 관계형 데이터 처리의 복잡한 조인 문제 대안으로 컬럼 지향적인 방식으로 데이터 구조를 세우는 시도가 있었으며, 이는 여러 테이블에 걸친 복잡한 조인에 대한 해결과 대용량 데이터에 대한 압축이 가능하게 해주는 장점이 있다[14,15].

3. 모바일 디바이스 UX 분석 프로세스

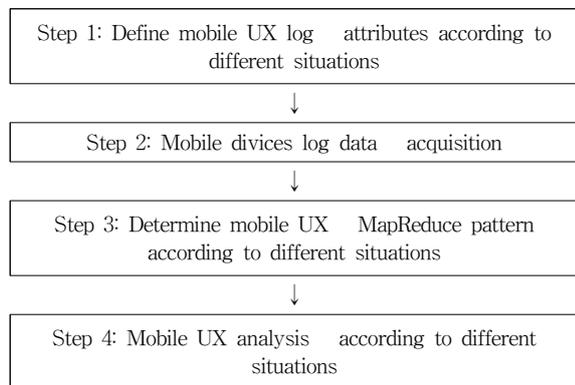


Fig. 1. Mobile device UX analysis process

모바일 디바이스 UX에 대한 분석을 위해 먼저 모바일 특성에 맞는 데이터 세트의 속성 정의 단계부터 시작하였다 (Fig. 1 참조). 모바일 환경에서는 하드웨어적인 기능과 소프트웨어적인 기능들을 고려하고 이에 반응하는 사용자의 상태와 동작 등도 반영되어야 하여, 이를 위한 데이터 세트에 대한 속성을 정의한 후 사용자들의 로그를 수집하기 위하여 수집기를 설치하였다.

MapReduce 패턴 설계 단계에서는 단순한 Map 과 Reduce의 조합이 아닌 처리하려는 모바일 데이터 크기나 속성에 맞게 하둠에서 제공하는 다양한 함수들을 사용하여 처리한다. 이때 분산처리에 적합한 다양한 배치를 고려하였다.

하둠은 Map과 Reduce 외에 데이터 분할과 Shuffling처럼 프레임워크가 제공하는 다른 기능들을 가지고 있다. 또한

Partitioning과 Combining 같은 기능을 수정해 사용할 수 있다. 따라서 대용량 데이터 처리 시 사용자는 서로 다른 옵션을 이용해 데이터를 다양하게 읽을 수 있으며 제공되는 기능들을 다양하게 조합하여 데이터를 처리할 수 있다.

분석 프로세스에서는 수집된 데이터를 분석하기에 앞서 이와 같이 하둠 프레임워크에서 제공해주는 다양한 기능들을 조합하여 모바일 UX 특성을 고려한 MapReduce 패턴에 대한 설계 단계를 포함하였다. 이에 따른 분석 프로세스는 다음과 같다.

3.1 모바일 데이터 세트 속성 정의 단계

모바일 디바이스의 UX 분석을 위한 대용량 데이터 처리 분석 프로세스에서는 모바일 환경에서 발생하는 하드웨어적인 상황과 소프트웨어적인 상황이 뒤섞여 있기 때문에 데이터를 수집할 때부터 처리하고자 하는 데이터의 속성들을 구분하여 데이터 세트를 기술해야 한다. Table 1에 본 연구에서 정의한 데이터 세트의 속성들을 나타내었다.

Table 1. The characteristics and definition of mobile user data set

Definition	Field
Time	Date
	Week
	Time
Mobile Device Keys	Key1
	Key2
Mobile UX Status	PackageName1
	PackageName2
	Event1
	Event2
	Status
	Description

3.2 모바일 디바이스 UX 로그데이터 수집 단계

모바일 디바이스 UX의 특성에 따라 정의된 속성의 값들을 로그 수집 프로그램에 장착하여, 사용자 로그를 수집한다. 모바일 디바이스 사용자의 하드웨어적인 버튼 사용 내역과 모바일 서비스를 통한 애플리케이션 사용에 대한 소프트웨어적인 속성이 기록된 수집 로그는 Fig. 2와 같은 형태를 보여준다.

```

    Time      Mobile Device Keys      Mobile UX Status
    ┌────────┬────────────────────────────────┬────────────────────────────────┐
    2013 | Jan | Sat | ##### | . | . | Android System [PatternUnlockScreen] | START | . | . | . | . |
    2013 | Jan | Sat | ##### | . | . | System_Dialogs_Status | System Dialogs Close | . | . | . | . |
    2013 | Jan | Sat | ##### | . | . | Battery_Status | Status: Discharging Health: Good Capacity: Li-ion | . | . | . | . | Li-ion Voltage: .... |
    2013 | Jan | Sat | ##### | . | . | Scaling_Status | Current: 1200000 | . | . | . | . | Scaling Range: .... |
    2013 | Jan | Sat | ##### | . | . | System_Disk_Status | 99% Used | . | . | . | . | Total: 516,040K Used: 480,032K Available: 36,008K |
    2013 | Jan | Sat | ##### | . | . | Data_Disk_Status | 38.4% Used | . | . | . | . | Total: 2,064,208K Used: 792,792K Available: 1,271,416K |
    2013 | Jan | Sat | ##### | . | . | Wi-Fi_Status | Link speed: -1 | . | . | . | . | SSID: <none>, BSSID: <none>, MAC: <none>, Supplicant state: .... |
    2013 | Jan | Sat | ##### | DEVICE_KEY | POWER KEY | Android System [PatternUnlockScreen] | . | . | . | . | . |
    2013 | Jan | Sat | ##### | DEVICE_KEY | HOME KEY | Android System [PatternUnlockScreen] | . | . | . | . | . |
    2013 | Jan | Sat | ##### | . | . | Dial | [ToggleButton] | . | . | . | . | VOLUME_VOICE [VALUE= 5] |
    2013 | Jan | Sat | ##### | . | . | Call_Status | The line is busy | . | . | . | . |
    
```

Fig. 2. The sample of mobile user log data

3.3 모바일 UX를 고려한 MapReduce 패턴 설계단계

분석하려는 데이터 셋에 대한 MapReduce 분산 처리를 위해서는 하둡에서 제공하는 Map과 Reduce 외의 다양한 기능들을 조합하여 어떻게 프로그래밍을 할 것인가에 대해 설계하는 단계가 필요하다. MapReduce로 처리하는 여러 프로그래밍 패턴 중 본 연구에서 사용한 패턴은 Numerical Summarization Pattern이다. 이는 대용량의 입력 데이터에 대하여 그룹 별로 사용빈도에 대한 카운팅을 하여 결과 값을 대폭 축소할 수 있는 기본적인 MapReduce 패턴이다. 이는 주로 Word Count, Record Count, Min/Max Count, Average/Median/Standard Deviation 등의 계산에 사용한다. Numerical Summarization Pattern의 구조는 다음과 같다[3].

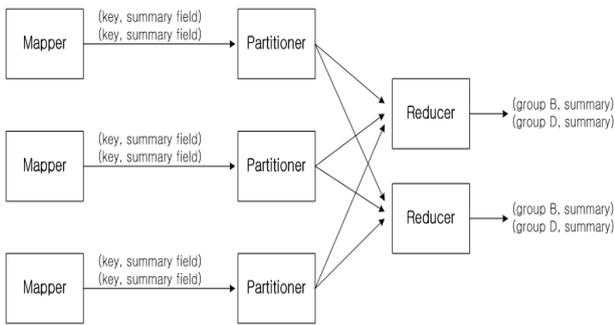


Fig. 3. The structure of numerical summarization pattern

분석에서 사용하는 Mapper는 표준 입력을 통해 들어온 스트림을 분할 데이터 셋으로 읽어 들이고 지시한 키에 따라 출력 값을 추출한다. 본 분석에서의 Mapper는 대용량의 전체 데이터 스트림을 입력으로 하고 레코드 단위로 분할한다. 즉, 반복적으로 한 번에 하나의 라인 별로 레코드를 읽고, 이렇게 읽어 들인 레코드를 구분자에 의해 필드별로 구분한다. 여기서 구분자는 “|”를 사용한다. 분석 플랫폼에서 사용한 패턴은 Combiner 기능 없이 Partitioner를 사용하여 Shuffle과정 없이 Reducer의 입력 값으로 바로 이동한다. 즉, Reduce 에서 같은 키에 대한 값의 리스트로 묶인 데이터들은 바로 연산되어 처리한다.

3.4 모바일 UX 데이터 분산처리 단계

모바일 UX 분석의 마지막 단계인 분산처리 단계에서는 정의된 속성 설계에 의하여 수집된 사용자 로그 데이터가 세 번째 단계에서 선택된 프로그래밍 패턴에 의하여 실제적으로 Map 태스크와 Reduce 태스크를 거치면서 분산처리 되는 단계이다. 두 번째 단계의 모바일 디바이스에서 수집한 대용량의 로그 데이터가 전체 프로세스의 세 번째 단계에서 선택한 MapReduce 설계 패턴의 입력 값으로 들어간다.

하둡에서 입력 데이터는 여러 개로 분할되어 분할된 일부 데이터를 대상으로 여러 노드에서 동시에 Map 태스크가 수행된다. 각각의 Map 태스크는 자신에게 할당된 입력 데이터를 대상으로 처리한 결과를 각 노드의 로컬 파일 시스템에 저장한다. Reduce 태스크는 여러 노드에 저장되어 있는

중간 결과를 입력으로 받아 통합 처리하여 최종 결과를 제공한다. 태스크 분배는 네트워크 트래픽 발생을 최소화하기 위해 데이터가 위치한 노드에서 수행하도록 분배되며, 하둡에서 제공하는 각 단계별 컴포넌트는 Table 2와 같다.

Table 2. The components of mapreduce tasks

Task	Component
Map Task	Record Reader
	Map
	Combiner
	Partitioner
Reduce Task	Shuffle and Sort
	Reduce
	Output Format

4. 분석 결과

본 논문의 모바일 로그 데이터 분석에서 사용한 MapReduce 프레임워크 버전은 아파치 하둡 1.0.3 이며, 우분투 리눅스를 실행 운영체제로 사용했다. 모바일 디바이스 UX의 특성에 따라 정의된 속성 값들을 로그 수집 프로그램에 장착하여 4주간(24시간/일, 20명)의 사용자 모바일 디바이스의 대용량 로그 기록을 수집하였으며, Numerical Summarization Pattern 구조를 적용한 MapReduce 수행을 통해, 다음과 같은 분석 결과를 얻을 수 있었다.

4.1 모바일 디바이스 사용 평일 주말 비교

모바일 디바이스의 스크린 온 상태의 시간을 모두 카운트한 결과 아래 그림에서 주 중에는 평일 평균 114분을 사용하였고 주말에는 평균 83분을 사용한 것을 알 수 있었다.

또한 주말 주중 모두 출근 시간대인 오전 7시부터 사용증가가 가파른 것을 확인할 수 있었다. 주중 퇴근 시간대인 오후 6시 경에 가장 많은 사용량을 나타냈다. 전체적으로 모바일 디바이스 이용이 주말보다는 주중에 더 많이 이루어지고 있음을 알 수 있다.

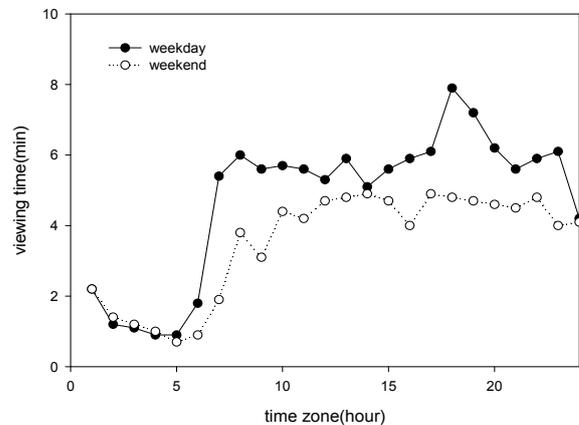


Fig. 4. The comparison of mobile device usage time between weekday and weekend

4.2 모바일 디바이스를 통한 시간대별 인터넷 이용

입력된 데이터 셋에서 각 시간대별로 모바일 디바이스를 통한 인터넷 사용량을 분석한 결과 다음과 같다. 그래프를 보면 오후 3시부터 밤 12시까지 인터넷을 가장 많이 사용하는 행태임을 알 수 있었다. 출근 시간대인 8시가 주위 시간대에 비해 사용빈도가 높고 저녁 시간대에서는 7시와 11시가 주위 시간대에 비해 사용빈도가 높게 나타난다. 이를 통해 사용자 인터페이스 화면 디자인에서 사용 시간대의 선호도에 따른 애플리케이션 개발을 유도할 수 있다.

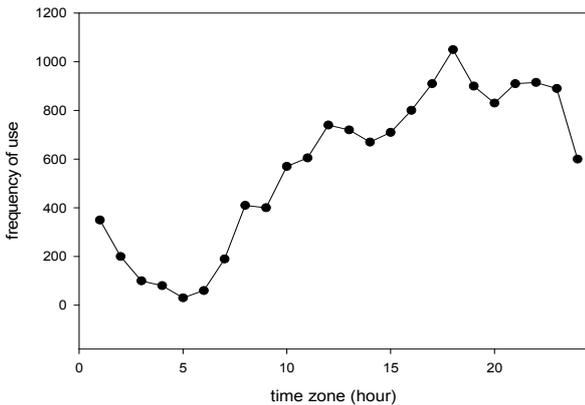


Fig. 5. Internet connection analysis according to time window

4.3 모바일 디바이스를 통한 시간대별 Application 사용

모바일 디바이스의 사용 애플리케이션 중에서 카카오톡 [16]에 대한 사용량을 분석한 결과 다음과 같다. 그래프를 보면 애플리케이션 특성 상 출/퇴근 시간대에 급격한 사용량의 증가를 보이고 있다. 또한, 늦은 저녁 시간대에 고르게 사용한 특성도 나타내고 있다. 이는 모바일 애플리케이션의 사용자에 대한 시간대 별 다양한 피드백 제공시 참고할 수 있다.

4.4 모바일 디바이스 사용키 결과

모바일 디바이스의 UX 분석의 하드웨어적인 특성을 고려한 사용키에 대한 빈도 분석 결과는 다음과 같다. 디바이스 사용 버튼 중 BACK_KEY에 대한 사용이 월등히 높음을

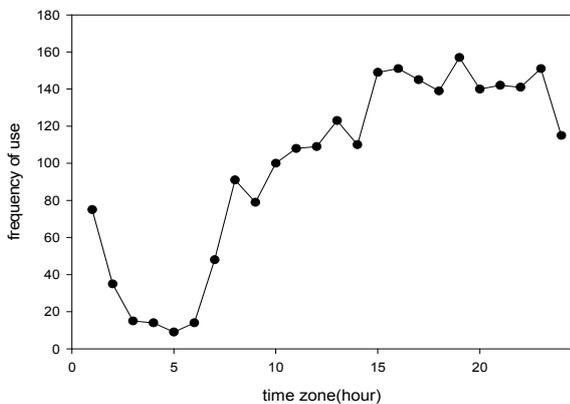


Fig. 6. Kakao Talk usage analysis according to time window

알 수 있다.

이를 통해 모바일 디바이스 하드웨어 버튼의 배치 위치에 대한 사용성 평가(Usability Analysis)[17]에서 BACK_KEY의 다른 키들과의 상당한 차이점을 보임에 따라 사용자의 선호 위치에 대한 요인 분석과 사용 빈도가 적은 VOLUME_KEY와 HOME_KEY에 대한 재배치 위치를 제안할 수 있었다.

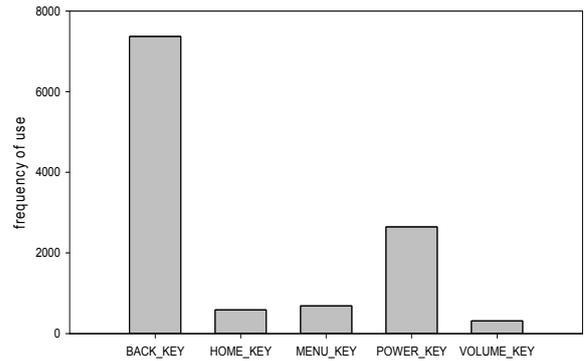


Fig. 7. Mobile device hardware key usage analysis

5. 결 론

다양한 모바일 디바이스에서 발생하는 대용량의 복합적인 로그 데이터는 데이터 일관성을 엄격히 요구하는 기존의 RDBMS의 데이터 모델 저장 방식으로는 분석에 한계가 있다. 이에 반해, MapReduce는 데이터의 확장성을 보장하는 키/값 방식, 반구조적 방식, 컬럼 지향 방식, 문서 지향 방식 등의 다양한 시도를 가능하게 한다.

모바일 디바이스의 UX 분석을 위하여 본 연구에서는 모바일 환경 변화를 고려하여 디바이스의 하드웨어적인 특성과 소프트웨어적인 특성을 고려하여 로그 속성을 먼저 정의하였으며, 정의된 속성에 따라 수집된 대용량의 로그 데이터는 본 연구에서 사용한 MapReduce 데이터 처리 방식의 Numerical Summarization Pattern 구조로 분산처리 하였다.

향후 관련 연구는 모바일 디바이스의 UX 분석을 위하여 다양한 형태의 모바일 디바이스의 특성에 따라 하드웨어적인 특성과 소프트웨어적인 특성의 좀 더 세분화된 속성 파악이 필요하다. 또한, MapReduce가 기본적으로 제공해주는 패턴을 벗어나 직접 분석하고자하는 데이터의 타입에 대한 조사와 이를 분산 플랫폼 환경에서 효율적으로 처리할 수 있는 다양한 MapReduce 프로그래밍 패턴에 대한 연구가 더욱 요청된다.

참 고 문 헌

[1] N. Laptev, K. Zeng, and C. Zaniolo, "Very Fast Estimation for Result and Accuracy of Big Data Analytics: the EARL System," in *Proceedings of the 29th IEEE International*

Conference on Data Engineering, Brisbane, 2013, pp.1296-1299.

[2] A. Balmin, T. Kaldewey, and S. Tata, "Clydesdale: Structured Data Processing on Hadoop," in *Proceedings of the ACM SIGMOD International Conference on Management of Data*, 2012, pp.705-708.

[3] D. Miner and A. Shook, "MapReduce Design Patterns," O'Reilly Media, ch. 2, pp.13-32, 2012.

[4] Y. S. Kim, "Secrets of Seductive UX/UI Design," Wiki Books, ch. 1, 2013.

[5] S. S. Kim, and S. G. Kim, "On Designing Cloud Computing Systems for Analysing Smart Device's UX," in *Proceedings of the 2013 Korea Computer Congress*, 2013, pp.303-305.

[6] S. Ghemawat, H. Gobiuff, et al., "The Google file system," in *Proceedings of the 19th ACM symposium on Operating systems principles*, 2003, pp.29-43.

[7] J. Dean and S. Ghemawat, "MapReduce: Simplified Data Processing on Large Clusters," in *Proceedings of the 6th Symposium on Operating System Design and Implementation, SanFrancisco*, 2004, pp.137-150.

[8] T. White, "Hadoop: the definitive guide," O'Reilly Media, 2009.

[9] J. Dean and S. Ghemawat, "MapReduce: A Flexible Data Processing Tool," *Communications of the ACM*, Vol.53, No.1, pp.72-77, 2010.

[10] M. Sethi, N. Sachindran, and S. Raghavan, "SASH: Enabling Continuous Incremental Analytic Workflows on Hadoop," in *Proceedings of the IEEE 29th International Conference on Data Engineering*, Brisbane, 2013, pp.1219-1230.

[11] F. Wang, J. Qiu, J. Yang, B. Dong, X. Li, and Y. Li, "Hadoop high availability through metadata replication," in *Proceedings of the 1st International Workshop on Cloud Data Management*, New York, 2009, pp.37-44.

[12] M. Y. Eltabakh, Y. Tian, F. Ozcan, R. Gemulla, A. Krettek, J. McPherson, "CoHadoop: flexible data placement and its exploitation in Hadoop," *The VLDB Endowment*, Vol.4, No.9, pp.575-585, 2011.

[13] A. Abouzeid, K. B. Pawlikowski, D. Abadi, A. Silbeschatz, and A. Rasin, "HadoopDB: An Architectural Hybrid of MapReduce and DBMS Technologies for Analytical Workloads," *The VLDB Endowment*, Vol.2, No.1, pp.922-933, 2009.

[14] Y. He, R. Lee, Y. Huai, Z. Shao, et al., "RCFile: A fast and space-efficient data placement structure in MapReduce-based warehouse systems," in *Proceedings of the 27th IEEE International Conference on Data Engineering*, Hannover, 2011, pp.1199-1208.

[15] Y. Lin, D. Agrawal, C. Chen, B. C. Ooi, S. Wu, "Llama: leveraging columnar storage for scalable join processing in the MapReduce framework," in *Proceedings of the 2011 ACM SIGMOD International Conference on Management of data*, New York, 2011, pp.961-972.

[16] KakaoTalk Enjoy free one-to-one or group chats anywhere in the world [Internet], <http://www.kakao.com/talk/ko>

[17] B. Albert, T. Tullis and D. Tedesco, "Beyond The Usability Lab: Conducting Large-Scale User Experience Studies," Morgan Kaufmann Pub., ch. 1, pp.5-16, 2010.



김 성 숙

e-mail : sungsook87@anyang.ac.kr
 1991년 한양대학교 교육공학과(학사)
 2009년 아주대학교 E-Learning학과(석사)
 2009년~현 재 안양대학교 컴퓨터공학과
 박사과정
 관심분야 : Big Data Analytics & Smart Learning



김 성 규

e-mail : sgkim@anyang.ac.kr
 1985년 홍익대학교 전자계산학과(학사)
 1987년 서울대학교 계산통계학과(석사)
 1994년 노스웨스턴대학교 컴퓨터과학과
 (박사)
 1987년~1996년 삼성종합기술원 정보시스템
 연구소 선임연구원
 1996년~현 재 안양대학교 컴퓨터공학과 교수
 관심분야 : Database and Knowledge base