韓國컴퓨터情報學會 論文誌
第16卷 第8號, 2011. 8.

2011-16-8-1-1

데이터 마이닝을 적용한 기업형 클라우드 컴퓨팅 기반 데이터 처리 기법

강 인 성*, 김 태 호*, 이 홍 철**

Data processing techniques applying data mining based on enterprise cloud computing

InSeong Kang*, TaeHo Kim*, HongChul Lee**

요 약

최근 클라우드 컴퓨팅은 인터넷 접속을 통해 언제 어디서든 사용할 수 있는 높은 이용편리성과 동시에 스마트폰, 넷북, PDA 등과 같은 각종 정보통신 기기로 데이터를 손쉽게 공유할 수 있는 사용환경을 제공하기 때문에 산업적 파급효과가 커 디지털혁명을 주도할 서비스로 주목받고 있다. 이와 같은 클라우드 컴퓨팅 기반의 협업 시스템을 통해 비즈니스 실무부서 간의 업무 통합이 점차적으로 이루어지면, 관련 부서 간 공유하게 되는 데이터가 더욱 많아지기 때문에 실무자가 필요한 데이터를 보다 쉽게 찾아 사용할 수 있는 방법이 필요하다. 기존 연구에서는 군집화를 통해 탐색과정을 단순화했지만, 본 논문에서는 관련 부서 간에 자주 발생하는 데이터 중복을 제거하고 시스템 성능을 향상시키기 위해 해쉬함수를 사용하고, 변경된 데이터에 대한 정보가 동적으로 반영되어 실무자에게 적합한 데이터가 분류될 수 있도록 데이터 마이닝 기법 중 베이지안 네트워크를 사용한 시스템을 제안하였다. 본 시스템은 기존 방법과 비교하여 탐색기능이 향상된 결과를 나타내었을 뿐만 아니라, CPU, Network Bandwidth 사용량 등의 시스템 성능에도 효율적인 것을 확인하였다.

▸ Keyword : 클라우드 컴퓨팅, 해쉬함수, 베이지안 네트워크

Abstract

Recently, cloud computing which has provided enabling convenience that users can connect from anywhere and user friendly environment that offers on-demand network access to a shared pool of configurable computing resources such as smart-phones, net-books and PDA etc, is to be watched as a service that leads the digital

• 제1저자 : 강인성 • 교신저자 : 김태호 • 책임저자 : 이홍철
• 투고일 : 2011. 03. 21, 심사일 : 2011. 04. 16, 게재확정일 : 2011. 05. 06.
* 고려대학교 정보경영공학부(Dept. of Information Management Engineering, Korea University)
** 고려대학교 산업경영공학부 교수(Dept. of Industrial Management Engineering, Korea University)

revolution. Now, when business practices between departments being integrated through a cooperating system such as cloud computing, data streaming between departments is getting enormous and then it is inevitably necessary to find the solution that person in charge and find data they need. In previous studies the clustering simplifies the search process, but in this paper, it applies Hash Function to remove the de-duplicates in large amount of data in business firms. Also, it applies Bayesian Network of data mining for classifying the respect data and presents handling cloud computing based data. This system features improved search performance as well as the results Compared with conventional methods and CPU, Network Bandwidth Usage in such an efficient system performance is achieved.

▸ Keyword : Cloud computing, Hash function, Bayesian network

I. 서 론

클라우드 컴퓨팅은 인터넷상의 서버를 통하여 데이터 저장, 네트워크, 콘텐츠 사용 등 IT 관련 서비스를 한 번에 사용할 수 있는 컴퓨팅 환경이다[1]. 최근 클라우드 컴퓨팅은 인터넷 접속을 통해 언제 어디서든 사용할 수 있는 높은 이용편리성과 동시에 스마트폰, 넷북, PDA 등과 같은 각종 정보통신 기기로 데이터를 손쉽게 공유할 수 있는 사용환경을 제공하기 때문에 산업적 파급효과가 커 디지털혁명을 주도할 서비스로 주목받으면서 차세대 패러다임으로 자리매김하고 있다[2]. 클라우드 컴퓨팅은 컴퓨팅 자산의 소유권, 물리적 위치, 대상 사용자의 범위 등을 고려하여 다양한 배치 방식으로 분류될 수 있고, 비교적 그 의미가 분명하고 널리 통용되고 있는 두 가지 배치 형태으로 기업형(사설) 클라우드 컴퓨팅(Private Cloud computing)과 공용 클라우드 컴퓨팅(Public Cloud computing)으로 나눌 수 있다. 특히 클라우드 컴퓨팅 시장은 확대되는 과정에서 데이터 보안에 대한 우려로 인해 기업형(사설) 클라우드 컴퓨팅 서비스가 주도적인 역할을 할 것으로 전망된다[3].

클라우드 컴퓨팅을 도입하면 기업 또는 개인은 PC에 자료를 보관할 경우 하드디스크 장애 등으로 인하여 자료가 손실될 수도 있지만 클라우드 컴퓨팅 환경에서는 별도 서버에 자료들이 저장되기 때문에 안전하게 자료를 보관할 수 있고, 저장 공간의 제약도 극복할 수 있으며, 언제 어디서든 자신이 작업한 문서 등을 열람·수정할 수 있다. 하지만 서버가 해킹당할 경우 개인정보가 유출될 수 있고, 서버 장애가 발생하면 자료 이용이 불가능하다는 단점도 있다[1]. 이 점 때문에 클라우드 컴퓨팅은 보안과 표준에 대한 연구가 주를 이루고 있다.

최근 클라우드 컴퓨팅의 상시적인 협업 체계 구축을 통한 업무 효율성 및 생산성 향상이 강조되고 있다[3]. 특히 클라우드 컴퓨팅 시장에서 주도적인 역할을 할 것으로 전망되는 기업형(사설) 클라우드 컴퓨팅 서비스를 통해 점차적으로 IT 부서와 비즈니스 실무부서의 차이가 점점 사라지고 있는 가운데, 관련 데이터 용량도 점점 방대해지고 있다[4]. 이에 따라 보안과 표준이 이루어진 가운데, 기업 내 실무자들의 작업능률 및 시스템 성능 향상을 위해 적시적소에서 필요한 데이터를 보다 쉽고 정확하게 분류하여 사용할 수 있는 연구가 절실히 필요하다.

본 논문에서는 관련 부서 간에 자주 발생하는 데이터 중복을 제거하고 시스템 성능을 향상시키기 위해 해쉬함수를 사용하고, 변경된 데이터에 대한 정보가 동적으로 반영되어 실무자에게 적합한 데이터가 분류될 수 있도록 데이터 마이닝 기법 중 베이지안 네트워크를 사용한 시스템을 제안하였다. 본 논문의 구성은 다음과 같다. 2장에서는 클라우드 컴퓨팅, 해쉬함수, 베이지안 네트워크, 협업 기반 추천시스템 등에 관한 개념 및 기존연구 등 관련연구에 대하여 알아본다. 3장에서는 시스템 모델링 및 테스트 대안 선정 및 시나리오 등 설계 방법을 제시하고, 종속변수, 설계변수 설정 및 시스템 비교 및 분석, 시스템 성능 평가 등 구현을 실시한다. 4장에서는 본 논문의 결론 및 향후 연구과제에 대한 내용으로 마무리하였다.

II. 관련 연구

1. 클라우드 컴퓨팅

1.1 개념 및 구성요소

클라우드 컴퓨팅은 그림 1과 같이 정보가 인터넷 상의 서버에 저장되고, 데스크톱·태블릿컴퓨터·노트북·넷북·스마트폰 등의 IT 기기 등과 같은 클라이언트에는 일시적으로 보관되는 컴퓨터 환경을 뜻한다. 즉 이용자의 모든 정보를 인터넷 상의 서버에 저장하고, 이 정보를 각종 IT 기기를 통하여 언제 어디서든 이용할 수 있다는 개념이다[1].

그림 1. 클라우드 컴퓨팅 개념
Fig. 1. The Concept of Cloud Computing

1.2 기존연구

클라우드 컴퓨팅에 관한 연구는 보안 인증 및 표준에 대한 연구가 주를 이루고 있지만, 최근 들어 리소스 할당 및 대규모 데이터 처리에 대한 연구도 진행 중이다.

정윤수(2011) 등은 특정 서버에 존재하는 데이터를 서로 다른 물리적인 위치에 존재하는 사용자가 제공받고 있는 경우, 임의의 사용자가 원격에서 특정 서버가 제공하는 데이터의 무결성 및 서버의 안전한 인증을 보장받기 위한 이중 해쉬 체인 기반의 플로딩 패킷 인증 메커니즘을 제안하였다. 제안된 인증 메커니즘은 특정 서버에 존재하는 데이터를 임의의 사용자가 안전하게 사용하기 위해서 특정 서버에 존재하는 $K(\geq 1)$개의 링크 상태에 대한 정보를 안전하게 사용자에게 플로딩시킨다. 또한 제안된 인증 메커니즘은 클라우드 컴퓨팅 환경에 적합하도록 사용자의 요구가 있을 경우에만 적당한 해쉬 값을 전달함으로써 중앙서버의 오버헤드를 낮추었다[5]. 한승민(2009) 등은 클라우드 마켓을 위해 QoS 및 S-Rank 분석을 이용하여 각 사용자가 원하는 상황에서 효율적인 리소스를 추천해주는 시스템을 제안하였다. 리소스를 등록하기 위해 각 서비스마다 QoS를 분석하고, 랜덤하게 서비스를 수행하는 시간과 라운드 로빈 방식을 이용한 서비스 수행시간, S-Rank를 이용한 서비스 수행 시간을 비교를 통해 클라우드 마켓의 활성화를 통해 컴퓨팅 효율을 증가시키며 더불어 에너지 효율도 높였다[6]. 전자통신동향분석(2009)은 기존의 RDBMS 기술, MPI 분산 처리 기술 등은 적용하기에는 운영 환경, 기능/성능 면에서 확장성 혹은 고비용 문제가 발생하기 때문에 대규모 클러스터 환경을 기반으로 한 시스템이 비즈니스 모델로 제시됨에 따른 데이터 저장 및 관리에 관한 기술 동향 및 발전 방향을 서술하였다.[7].

2. 해쉬함수

2.1 개념 및 구성요소

해쉬함수는 요약함수라고도 하며, 주어진 원문에서 고정된 길이의 난수를 생성하는 연산기법이다. 이 때 생성된 값은 '해시값'이라고 한다. Chord[8], Pastiche[9] 같은 P2P 오버레이 네트워크에서 라우팅을 위해 그리고 파일의 중복 저장을 막기 위해 MD5[10], SHA1[11] 해시 함수를 사용하고, 특히 파일 저장 시 파일의 해시를 만들고 서로 비교하여 같은 해시이면 중복으로 처리하여 저장을 막는다.

데이터를 바이트 단위로 비교하여 중복 데이터를 찾는 일은 많은 시간을 소모한다. 데이터가 많을수록 파일의 개수가 증가 하는데 파일의 개수를 n이라고 했을 경우 비교 횟수가 n(n+1)/2로 나타나므로 n이 커질수록 비교 횟수가 점점 증가한다. 또한 바이트 단위의 비교는 액세스 속도가 느린 디스크의 빈번한 입출력을 유발시키기 때문에 오버헤드가 크다. 그러나 직접적인 바이트 비교 대신 해시 함수를 사용한다면 큰 오버헤드 없이 중복 데이터를 찾을 수 있다. 일정한 크기의 데이터를 MD5, SHA1과 같은 해시 함수를 사용하면 해시값(checksum)이 생성되는데, 해시값이 서로 같다면 동일한 데이터로 보는 것이다. 해시 함수의 충돌 확률을 구하는 식은 식 1과 같다. n이 입력 데이터의 양이고 k가 해시값이 표현할 수 있는 범위이다. 만약 국가적인 사업을 위해 Exabyte를 담을 수 있는 저장이 필요하고 해시 함수에 입력되는 평균 크기를 8Kbyte라고 가정해보자. 이때 사용하는 해시 함수로 SHA1를 사용한다면 입력 데이터는 $2^{60}/2^{13}$, 즉 n은 2^{47}로 표현할 수 있고 k는 SHA1 해시값의 크기 160bit로 설정하여 계산하면 해시 충돌이 일어날 확률이 약 10^{-17}로 계산되며, 발생확률이 매우 작음을 확인할 수 있다[12].

식 1. 해시 충돌 확률
Formula 1. The Probability of Hash Crash

$$1-\frac{n!}{(n-k)!n^k} > 1-e^{-\frac{k(k-1)}{2n}}$$

$$k=160,\ prob=10^{-17}$$

2.2 기존연구

해쉬함수에 관한 연구는 상호 인증에 대한 연구가 주를 이루고 있으나, 최근 데이터 중복에 대한 연구를 통해 저장 효율을 향상시키려는 연구가 함께 진행되고 있다.

정호민(2010) 등은 해시를 사용하여 데이터 중복을 찾는 기법에 대한 기존 연구의 한계를 분석하고 다양한 파일 서버에서 범용적으로 사용 가능한 개선된 중복제거 알고리즘을 제시하였다. 고정 분할 기법에 스트라이드(stride) 기법을 적용하는 방식을 제안하여 중복 영역의 블록을 찾아내는 시간을 최소화하고 효율적으로 저장 시스템을 관리하는 것을 보였다[12].

이연(2006) 등은 데이터 웨어하우스에서 해쉬 테이블을 이용한 효율적인 데이터큐브 생성 기법을 제안한다. 제안 기법은 데이터큐브 생성 시 가중치 맵핑 테이블과 레코드 해쉬 테이블을 사용하여 다차원 데이터의 저장될 레코드 순서를 빠르게 찾아 저장한다. 따라서 데이터큐브의 생성속도가 향상되며 해쉬 테이블 만을 유지하여 메모리 사용량이 감소한다. 이는 성능평가를 통해 기존 기법보다 데이터의 빠른 검색과 데이터큐브 생성 요청에 빠른 응답을 보였다[13].

이규옥(2001) 등은 결합 이전에 어느 정도의 결합률을 예측할 수 있다는 전제하에 다중 해쉬 결합 실행 시에 발생할 수 있는 지연 시간을 최소화 할 수 있도록 결합률에 따라 최적의 프로세서들을 노드에 할당함으로서 다중 해쉬 결합의 실행 성능을 개선하였다. 그리고 분석적 비용 모형을 세워 기존 방식과의 다양한 성능 분석을 통해 비용 모형의 타당성을 입증하였다[14].

3. 베이지안 네트워크

3.1 개념 및 구성요소

베이지안 네트워크는 확률변수들 간의 의존관계를 네트워크 구조를 사용하여 문제의 대상을 표현하는 확률모델이다[15, 16]. 베이지안 네트워크는 실체(entity)들 간의 인과관계를 나타내는 확률 모델로서, 관측이 곤란한 요소를 다루는 것이 가능하고 추측되는 가설의 확신도를 실제 데이터를 바탕으로 검증을 할 수가 있으며, 전문가의 지식을 네크워크 구조로 표현하는 것이 가능한 장점이 있다. 변수는 노드로, 변수 간의 의존관계는 원인으로부터 결과가 되는 변수로서 방향을 가지는 유향링크로 그림 2와 같이 나타낼 수 있다[17].

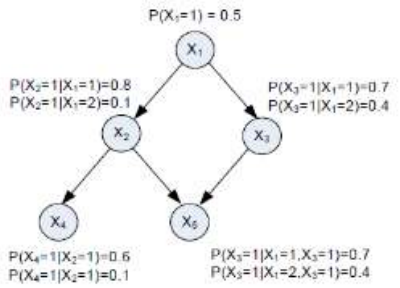


그림 2. 베이지안 네트워크 구조
Fig. 2. Bayesian Network Structure

3.2 기존연구

베이지안 네트워크에 관한 연구는 상태 추론 및 패턴 분석을 통한 시스템 성능 향상에 대한 연구가 주를 이루고 있다.

이유정(2006) 등은 베이지안 네트워크를 기반으로 생성하고 평가한 가상예제를 활용하여 범주 속성 데이터에 대한 분류 성능을 향상시키는 방안을 제안하였다. 스마트폰의 사용자 통신기록을 PC로 이동하여 PC에서 사례기반 추론을 이용한 친밀도 조정 기능을 수행함으로써 스마트폰의 제한된 계산능력 및 저장 용량을 해결할 수 있었다[18]. 황정식(2005) 등은 상품구매에 따르는 소비자의 구매행동 패턴을 분석하기 위해 판매자의 노하우와 소비자의 구매의식을 조사하여 이 데이터를 바탕으로 베이지안 네트워크를 구성하고 구매패턴을 예측하는 방법을 제안하였다. 판매자의 노하우와 소비자의 구매의식 조사 데이터를 바탕으로 베이지안 네트워크를 구성하였고 불필요한 속성을 가진 데이터를 제거하여 구매패턴을 분석하는데 정확도를 높일 수가 있었다[19]. 정경용(2003) 등은 기존의 사용자 선호도 예측 방법의 문제점을 보완하기 위하여 베이지안 추정치가 부여된 유사도 가중치와 연관 사용자 군집을 이용한 선호도 예측 시스템을 제안하였다. 추정치가 부여된 선호도를 기존의 피어슨 상관관계에 적용할 경우 결측치(Missing Value)로 인한 예측의 오류를 적게 하여 예측의 정확도를 높일 수 있다. 제안된 방법의 성능을 평가하기 위해서 기존의 협력적 필터링 기술과 비교 평가하였다[20]. 최준혁(2002) 등은 유사한 선호도를 보이는 사용자를 대상으로 군집분석을 수행함으로서, 이웃 사용자를 선택하는 과정을 단순화할 수 있고, 또한 베이지안 학습을 이용하여 사용자의 선호도를 동적으로 갱신할 수 있는 알고리즘을 설계하고 구현하였다. 이를 웹 도서 추천시스템에 적용하여 사용자의 만족도를 증가시킴을 보였다[21].

4. 협업 기반 추천시스템

협업 기반의 필터링을 이용한 추천 시스템은 대부분의 추천 시스템에서 사용하는 기술이다. 이는 서비스에 대한 사용자의 의견이 바로 반영되어 추천을 해주기 때문이다. 비슷한 성향(취미, 흥미, 관심 등)을 가지고 있는 사용자들로 분류 후 기존 사용자들의 평가들을 바탕으로 현재 사용자에게 원하는 정보를 추천하게 된다. 하지만 "Cold-Start"[22], "Data Sparseness"[23] 라는 문제점을 가지고 있다. "Cold-Start" 는 새로운 서비스가 들어오는 경우 기존의 유사한 속성을 지닌 서비스보다 높은 품질을 지니고 있지만 추천 시스템은 기존의 반영된 사용자의 의견을 기반으로 추천해 주므로 새로운 서비스에 대한 관심은 떨어지게 되어 발생하는 문제이고, "Data Sparseness"는 너무 많은 서비스에 대한 평가로 인하여 발생하는 문제이다.

III. 본 론

1. 시스템 설계

1.1 시스템 모델링

본 시스템은 그림 3과 같이 구성하였다. 기업은 실무 담당자들이 동 시간대에 엄청난 데이터를 공유하기 때문에 최소한의 데이터 처리가 절실히 필요하다. 본 시스템은 우선 각 사용자가 PC로 데이터를 업로드하도록 하였다. 현재 상용되는 시스템들은 대체적으로 동일한 파일명에 대해서만 중복 여부를 판단하지만, 본 시스템은 파일명은 다르더라도 동일한 내용의 데이터인지 여부를 우선적으로 해쉬함수를 통해 중복데이터를 제거하고 전송한다. 저장된 서버 내의 데이터들은 상시 협업을 위해 사용 시 데이터별로 부여된 부서 관련도 및 중요도를 바탕으로 베이지안 네트워크를 통해 우선순위를 할당하여 담당자의 실무에 가장 적합한 데이터를 제공하게 된다. 이 때, 서버 내에는 해쉬함수를 통해 동일 데이터는 존재하지 않고, 사용자의 권한은 모두 동일하여 모든 데이터 접근이 가능하다고 가정한다.

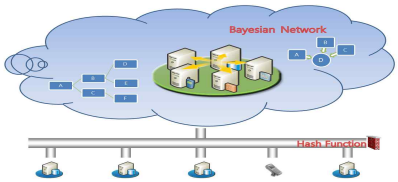


그림 3. 시스템 모델링
Fig. 3. System Modeling

본 시스템은 SERVER-PC간 전송을 통해 성능을 확인하며, 시스템 개발환경은 표 1과 같다. Client-PC의 경우 PC 수는 동일성능의 PC 10대(XP사용 7대, Linux사용 3대)를 사용하였다. 개발 간 Java기반으로 운용하여 OS상의 제약은 없었다.

표 1. 시스템 환경
Table 1. System Environment

H-W					
	CPU	RAM	HDD	LAN	OS
Server	i3-540	3G	500GB	100Mbps	XP
Client	P4-3.0	1G	350GB	100Mbps	XP, Linux
S-W					
Web Server	Apache Tomcat 7.0				
SQL	Mysql 5.5				
Developer	Eclipse EE for Java				
Analysis	BWmeter, Heavyload, Weka, SPSS				

1.2 테스트 대안 선정 및 시나리오

테스트를 위해 처리 기법 적용 전과 적용 후로 나누어 실시하였다. 처리 기법 적용 전은 중복된 데이터의 여부를 처리 없이 전송을 실시하고, 우선순위나 가중치를 부여하여 데이터를 분류하는 과정 없이 모든 데이터에 대한 조건이 동일한 상태로 시스템을 운영하였다. 처리 기법 적용 후는 데이터 전송 시 그림 4와 같이 해쉬함수를 통해 중복여부를 처리하여 데이터 중복을 제거하고, 그림 6과 같이 실무자가 설정한 관련도 및 중요도에 대한 데이터별 메타 데이터를 바탕으로 베이지안 네트워크를 통해 사용자에게 가장 적합한 데이터를 분류하여 출력하도록 하였다.

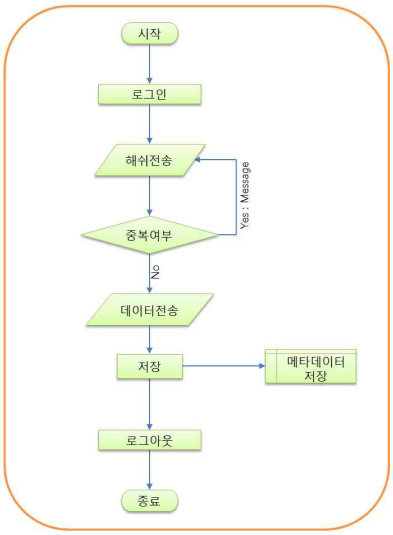


그림 4. 1단계 순서도
Fig. 4. Step1 Flow-Chart

Step1. 그림 4의 순서도 내용은 다음과 같다.

1) 그림 5와 같이 로그인을 실시하여 개별 사용자의 데이터 관리가 가능하고, 개인별 특성에 맞는 서비스가 가능토록 한다.

그림 5. 로그인 화면
Fig. 5. Log-in Page

2) 선택한 데이터를 전송한다.
3) 서버 내 해쉬함수 리스트와 비교를 위해 데이터의 해쉬를 생성하고 전송한다.
4) 중복여부를 판단하여 Yes시 중복에 대한 메시지를 전송하고, No시 선택한 데이터를 전송한다.
5) 데이터는 저장 시, 내부저장소에 데이터별 관련도, 중요도에 대한 메타데이터를 함께 저장하여 차후 우선순위 설정 및 분류에 사용가능토록 한다.

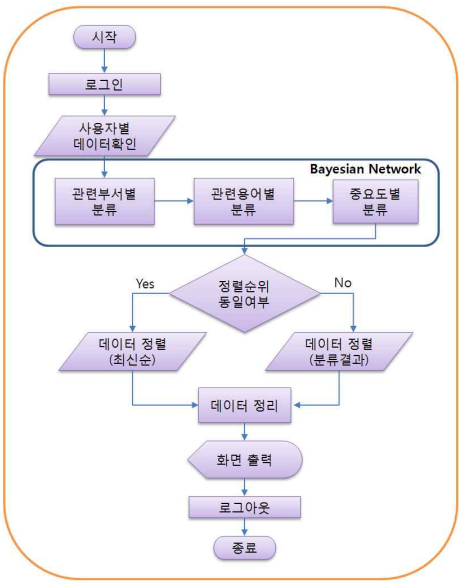


그림 6. 2단계 순서도
Fig. 6. Step2 Flow-Chart

Step2. 그림 6의 순서도 내용은 다음과 같다.
1) 로그인을 통해 사용자별 데이터를 확인한다.
2) 타 사용자의 데이터 중 사용자의 실무와 관련한 관련부서별, 관련용어별, 중요도별 순으로 베이지안 네트워크를 이용하여 데이터를 탐색한다.
3) 탐색한 결과를 바탕으로 정렬순위 동일여부를 파악하고, Yes시 최신 데이터부터 정렬하고, No시 분류된 결과대로 정렬한다.
4) 정렬된 타 데이터를 사용자의 본 데이터에 이어 차례대로 출력한다.

2. 시스템 구현

2.1 종속변수 설정

본 논문에서는 시스템의 종속 변수는 표 2와 같다.

표 2. 종속변수 설정
Table 2. Set Dependent Variable

	종속변수	단위
1	중복 데이터 제거율	%
2	데이터 분류 정확도	%
3	시스템 CPU 사용량	%

중복 데이터 제거율은 데이터 분류에 앞서 데이터 처리 및 전송 간 시스템에 미치는 영향을 확인할 수 있고, 데이터 분류 정확도는 현업의 실무특성 상 일정 시간 안에 작업이 시작하기 위해 정확한 데이터를 찾고 스케줄에 맞게 작업이 이루어질 수 있는가를 확인하기 위한 지표로 사용된다. 시스템 CPU 사용량은 대부분의 작업이 동 시간대에 이루어지고 이용률이 너무 높아져 사용이 지연하다면 시스템의 병목현상이 발생하기 때문에 종속변수로 설정하였다. 더 많은 종속 변수가 고려된다면, 민감도 분석을 통하여 종속 변수와 설계 변수의 선택도 가능할 것이다.

2.2 설계변수 설정

설계(독립) 변수를 선정하는 방법은 종속 변수에 가장 큰 영향을 주는 요소들을 선택하여야 한다. 하지만 어떤 변수가 종속 변수에 가장 큰 영향을 주는지 확신하지 못하는 상황이라면 시스템의 특성을 반영할 수 있는 모든 변수를 선정하게 되는데, 본 시스템의 특성상 해쉬함수로 인한 데이터 중복처리와 베이지안 네트워크를 통한 관련도 및 중요도에 대한 데이터 분류가 독립변수로 사용되는 것이 가장 효율적인 선정 방법이 될 것이다.

2.3 시스템 비교 및 분석

Java에서는 암호화 및 메시지 검증 코드를 구현해주는 클래스를 제공해준다. JCE(Java Cryptography Extension)란 프레임워크로 J2SE 1.4이후버전에 기본적으로 포함되어 있다. 시스템에 업로드되는 데이터는 표 3과 같이 Java를 이용한 Hash생성를 통해 각 데이터별 해쉬코드를 비교하고, 데이터가 중복되어 업로드되는 것을 방지하여 시스템 상에 중복 데이터가 존재하지 않도록 하였다.

표 3. Hash 생성 코드
Table 3. Generate Hash Code

```
import java.security.*;
..
byte[] bytesOfMessage
= yourString.getBytes("UTF-8");
MessageDigest md
= MessageDigest.getInstance("MD5");
byte[] thedigest = md.digest(bytesOfMessage);
```

관련도 및 중요도에 대한 변수를 n개의 확률변수 $x_1 \ldots . x_n$로 나타내어 베이지안 네트워크를 이루고 있는 노드라고 가정하고 조건부 독립이라는 가정 하에 이 네트워크의 모든 노드에 대한 결합 확률은 식 2와 같이 적용하였다. 만약 '연구개발' 부서의 '개발'관련 자료가 실무자에게 '보통'의 평가를 받고 업로드되면, 데이터를 사용할 실무자의 해당 부서별 데이터 관련도 순위에 할당된 확률(X_i)을 할당하고, 관련 데이터의 중요도에 할당된 확률(X_j)을 결합하여 적합한 데이터를 우선적으로 분류하게 된다.

식 2. 베이지안 네트워크 결합 확률
Formula 2. The Probability of Combination

$$P(X_{1,..}X_n) = \Pi P(X_j | P(X_i))$$

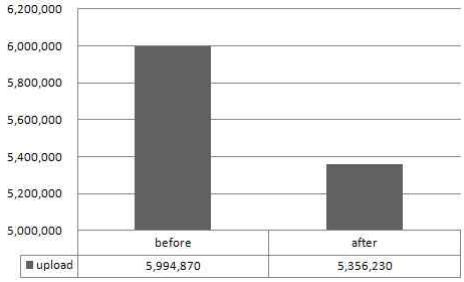
시스템의 데이터는 표 4와 같이 부서테이블, 관련분야 테이블을 기초코드로 하여 사용자별 부서별 관련분야 데이터와 데이터별 관련분야와 중요도에 대한 데이터로 나누었고, 데이터별 관련분야와 중요도는 Random Variable을 통해 생성하였다.

데이터 중복 제거로 인한 데이터 전송량 변화를 확인하게 위한 BWmeter[24]과 선정된 종속변수를 측정하고 비교하기 위해 베이지안 네트워크를 통한 분류 성능측정을 위한 Weka[25], CPU의 성능을 효과적으로 측정하여 그래프로 보여주는 HeavyLoad[26]을 사용하였다. 정확한 성능 변화를 확인하기 위해 표 5의 네트워크 사용량 변화와 표 6과 같이 베이지안 네트워크를 실행한 표 7의 분류결과, 그림 7, 8의 적용 전·후의 CPU사용량을 산출하였다.

2.4 시스템 성능 평가

제안된 시스템을 이용하여 각 종속변수에 따른 비교를 실시하고 결과를 산출하였다. 우선 자바를 통해 구현한 해쉬코드 생성을 통해 중복된 데이터를 찾아 제거하고, 네트워크 사용량을 측정한 결과는 표 5와 같이 네트워크 사용량 전후비교를 통해 업로드 용량이 다소 감소하는 것을 확인할 수 있었다.

표 5. 네트워크 사용량 비교 (단위 : Kilobyte)
Table 5. Measure Before Against After (Unit : Kilobyte)



중복이 제거된 데이터는 표 6과 같이 Weka를 통해 150건에 대해 Resampling을 실시하여 데이터를 전처리하고, Classify의 NaiveBayes를 통해 분류하였다. 베이지안 네트워크를 위한 데이터는 Nominal 데이터가 필요하기 때문에 분석 시에는 데이터를 각 코드별 데이터(예 : 부서테이블, 코드 1 = 기획)로 변경하여 실시하였다. 그 결과, 표 7과 같이 본 데이터가 88.67%의 분류성능을 보여주는 것을 확인할 수 있었다.

표 6. Weka 실행 결과
Table 6. Run Weka

```
=== Run information ===

Scheme:weka.classifiers.bayes.NaiveBayes
Relation:
2-weka.filters.unsupervised.attribute.Remove
-R2-5-
weka.filters.supervised.instance.Resample
-B0.0-S1-Z100.0
Instances:    150
Attributes:   5
```

표 4. 데이터 형식 예
Table 4. The Example of Data Form

부서	내용
1	기획
2	연구개발
3	기술
4	마케팅
5	해외
6	경영지원
7	재무회계
8	법무
9	운영
:	:

관련	내용
1	기획
2	연구
3	개발
4	기술
5	마케팅
6	해외
7	경영
8	지원
9	재무
:	:

중요	내용
1	불필요
2	다소중요
3	보통
4	중요
5	매우중요

사용자	부서	관련1	관련2	관련3
A	1	1	17	26
B	2	2	3	23
C	3	4	19	23
D	4	5	13	22
E	5	6	15	23
F	6	7	8	29
G	7	9	10	27
H	8	11	14	25
I	9	12	16	29
:	:	:	:	:

데이터	관련	중요	데이터	관련	중요
A1	24	1	H9	13	2
A2	24	3	H10	30	3
A3	27	1	I1	12	4
A4	8	2	I2	4	3
A5	3	2	I3	8	1
A6	8	5	I4	18	3
A7	19	5	I5	4	1
A8	22	2	I6	26	5
A9	5	5	I7	15	4
:	:	:	:	:	:

또한 시스템 상의 CPU 전후 사용량 비교를 분석해 보면 초기 검색 시 비슷한 모습을 그리지만, 그림 7과 같이 처리 기법 적용 전의 경우 추가적인 탐색이 발생되어, 40~50%의 사용량이 다시 발생한 반면, 그림 8과 같이 적용 후의 경우 안정된 사용이 진행되는 것을 확인하였고, 이는 적용 후 추가적인 탐색이 거의 필요하지 않기 때문이라고 분석하였다. 시스템 탐색에 따른 CPU사용량에 대해 비교 분석한 결과 처리 기법이 적용된 시스템이 보다 우수한 결과를 보여주고 있다.

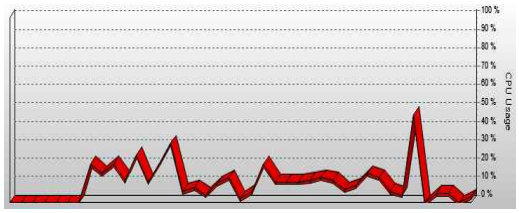


그림 7. CPU 사용량(적용 전)
Fig. 7. Using CPU(before)

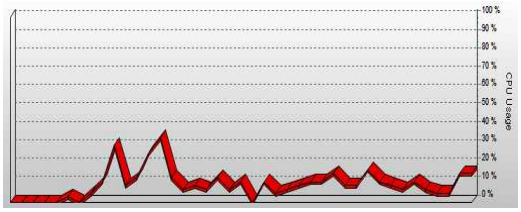
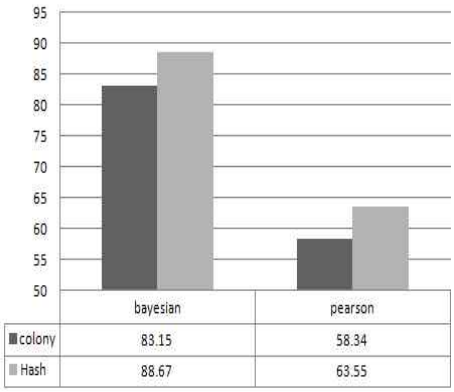


그림 8. CPU 사용량(적용 후)
Fig. 8. Using CPU(after)

기존 연구는 관련 아이템을 예측하기 위해 전체 사용자를 탐색 대상으로 설정하기보다는 사용자가 소속되어 있는 군집만 탐색하도록 함으로써 검색의 소요시간을 단축하기 위해 군집화를 수행하였지만, 본 연구에서는 해쉬함수를 통해 중복 데이터를 제거하여 기본적으로 대용량 데이터 시스템 적용에 따른 성능까지 고려하여 탐색성능향상을 위한 설계를 제시하였다.

기존 성능 비교 방법과 제시한 방법에 대한 성능 수준에 대해 평가하기 위해 유사한 데이터 분류에 관한 시스템에서 많이 사용되고 있는 피어슨의 상관계수 알고리즘과 비교하였다. 총 150건의 데이터를 이용하여 각각 해쉬함수를 통한 데이터중복 제거와 군집화를 우선 실시하여 데이터를 처리하고, 베이지안 네트워크와 피어슨 상관계수 알고리즘을 통해 성능을 평가하였으며, 두 방법 간의 탐색 성능 평가는 표 7과 같다. 표 7을 통해 해쉬함수를 통한 데이터를 이용하고 베이지안 네트워크를 통해 분류한 탐색이 88.67%로 기존의 군집화된 데이터에 대한 기존 탐색보다 성능이 향상되었음을 확인할 수 있었으며, 피어슨의 상관계수 알고리즘보다도 높게 나타났다.

표 7. 탐색 성능 평가 (단위 : %)
Table 7. Estimate Research Performance (Unit : %)



IV. 결 론

기존 실시했던 프로젝트를 통해 본 논문에서 제시한 기법을 적용하지 않은 일반적인 경우, 수많은 데이터 중에서 정확히 필요한 데이터를 선별하고 사용하기까지 불필요한 시간이 소요되고, 서버의 자원사용량이 추가적으로 발생하며, 동 시간 대 접속자가 많아질 경우 병목현상까지 발생할 수 있다는 점을 발견할 수 있었다. 이에 기업형(사설) 클라우드 컴퓨팅 기반 시스템과 같은 협업 데이터 처리를 위하여 관련 부서 간에 자주 발생하는 데이터 중복을 제거하고 시스템 성능을 향상시키기 위해 해쉬함수를 사용하고, 변경된 데이터에 대한 정보가 동적으로 반영되어 실무자에게 적합한 데이터가 분류될 수 있도록 데이터 마이닝 기법 중 베이지안 네트워크를 사용한 시스템을 제안하였다.

기업형(사설) 클라우드 컴퓨팅과 같은 협업에 적합한 시스템의 특성상 실무자가 동시간 대 동일 서비스공간에 공존할 수 있기 때문에 자원 할당량과 효율 향상이 매우 중요하다. 본 실험과 같이 자원 사용량이 감소하고, 탐색 시간이 줄어드는 것은 운영 프로세스 개선을 통한 실무자의 작업 능률 향상은 물론 설비의 유지 보수에도 큰 장점을 작용할 것으로 보인다. 또한 기존 방법과 성능을 비교하여 향상된 것을 확인할 수 있었다.

본 논문에서는 시스템 내 모든 사용자의 권한이 동일하고, 서버 내 모든 데이터에 대한 접근이 가능하다는 제안사항을 두고 있다. 따라서 협업을 위한 기업형 데이터 관리를 목적으로 한 실험이 가능할 수 있었으나, 본 논문에 관련하여 향후 과제로 클라우드 컴퓨팅 사용자 별 데이터 접근 권한과 이로 인한 보안문제 해결이 남아있다.

V. 감사의 글

이 연구에 참여한 연구자(의 일부)는 'BK21사업'의 지원비를 받았음.

참고문헌

[1] Cloud Computing, http://100.naver.com/
[2] Cloud Computing, http://terms.naver.com/
[3] Consortium of Cloud Computing Research http://www.cccr.or.kr/bin/
[4] Yun-Hee Lee, "Cloud Computing for Information Industry Business Innovation adoption," CIO Report, 2009.
[5] Yoon-Su Jeong, Yong-Tae Kim, "An Authentication and Integrity Guarantee Mechanism of Flooding Packet based on Double Hash Chain," Korean Institute of Information Technology, Vol. 9, No. 1, January 2011.
[6] Seung-Min Han, Eui-Nam Huh, Chang-woo Youn, "Efficient Resource Recommendation System for Cloud Market Computing," Korean Society for Internet Information, Vol. 11, No. 3, June 2010.
[7] Electronics and Telecommunications Research Analy sis," ETRI, 2009.
[8] Ion Stoica, Robert Morris, David Karger, M.Frans Kaashoek, and Hari Balakrishnan, "Chord: A Scalable Peer-to-peer Lookup Service for Internet Applications," ACM SIGCOMM 2001, San Deigo, CA, August 2001.
[9] L. P. Cox, C. D. Murray, and B. D. Noble. "Pastiche: Making backup cheap and easy," In Proc. 5th USENIX OSDI, Boston, MA, December 2002.
[10] R. L. Rivest, "The MD5 Message Digest Algorithm," Request for Comments(RFC) 1321, Internet Activities Board, 1992.
[11] RFC 3174, "US Secure Hash Algorithm 1 (SHA-1)"
[12] Ho-Min Jung, Young-Woong Ko, "Storage System Perfo rmance Enhancement Using Duplicated Data Management Scheme," Korea Institute of Information Scientists and Engineers, Vol. 37, No. 1, February 2010.
[13] Li Yan, Kim Hyung Sun, Byeong-Seob You, Jae-D ong Lee, Hae-Young Bae, "Data Cube Generation Method Using Hash Table in Spatial Data Warehouse," Korea Multimedia Society, Vol. 9, No. 11, November 2006.
[14] Kyu-Ock Lee, Man-Pyo Hong, "Efficient Processor Alloc ation based on Join Selectivity in Multiple Hash Joins using Synchronization of Page Execution Time,", Korea Institute of Information Scientists and Engineers, Vol. 28, No. 3,4, April 2001.
[15] David Heckerman, "A Tutorial on Learning Bayesia n Ne tworks," Technical Report MSR-TR -95-06, 1995.
[16] Thomas Dean et al, "Artificial Intelligence Theory an d Practice," Addison Wesley, 1995.
[17] Jensen, F. V., Bayesian Networks and Decision Gra phs, Springer-Verlag Berlin Heidelberg New York, 2001.
[18] Yujung Lee, Byoungho Kang, Jaeho Kang, Kwang ryel Ryu, "Generation and Selection of Nominal Virtual Examples for Improving the Classifier Performance," Korea Institute of Information Scientists and Engineers, Vol. 33, No.12, 2006.
[19] Jeong-Sik Hwang, Su-Young Pi, Chang-Sik Son, Hwa n-Mook Chung, "A Purchase Pattern Analysis Using Bayesian Network and Neural Network," Korea Intelligent Information Systems, Vol. 15, No. 3, June 2005.
[20] Kyung-Yong Jung, Seong-Yong Choi, Kee-Wook Rim, Jung-Hyun Lee, "Preference Prediction System using Similarity Weight granted Bayesian estimated value and Associative User Clustering," Korea Institute of Information Scientists and Engineers, Vol. 30, No. 3,4, April 2003.
[21] JunHyeog Choi, DaeSu Kim, KeeWook Rim, "Dyna mic Recommendation System for a Web Library by Using Cluster Analysis and Bayesian Learning," Korean Society for Internet Information, Vol. 12, No. 5, October 2002.
[22] J. Ben Schafer, Dan Frankowski, Jon Herlocker and Shilad Sen, "Collaborative Filtering Recommender Systems," The Adaptive Web, 2007.
[23] Gediminas Adomavicius and Alexander Tuzhilin, "Toward the Next Generation of Recommender Systems: A Survey of the State-of-the-Art and Possible Extensions," IEEE Educational Activities Department, 2005.
[24] BWMeter, http://www.desksoft.com/BWMeter.htm
[25] Weka, http://www.cs.waikato.ac.nz/ml/weka/
[26] Heavyload, https://www.jam-software.de/customer s/

저 자 소 개

강 인 성

2002 : 단국대학교 산업공학과 공학사.
현 재 : 고려대학교 정보경영공학과 석사과정
관심분야 : SI, SOA, SCM
Email : isk917@korea.ac.kr

김 태 호

2006 : 고려대학교 산업시스템정보공학과 공학사.
2008 : 고려대학교 산업시스템정보공학과 공학석사.
현 재 : 고려대학교 정보경영공학과 박사과정 수료
관심분야 : SI, SOA
Email : airth@korea.ac.kr

이 홍 철

1983 : 고려대학교 산업공학과 학사.
1988 : Univ. of Texas 산업공학 석사.
1993 : Texas A&M Univ. 산업공학 박사.
현 재 : 고려대학교산업경영공학부 교수
관심분야 : SCM, 생산 및 물류정보시스템
Email : hclee@korea.ac.kr